Developing hybrid frameworks for modelling reaction-diffusion systems (Alternative Format Thesis)

Smith, Cameron

Award date:
2021

Awarding institution:
University of Bath

Developing hybrid frameworks for modelling reaction-diffusion systems

submitted by

Cameron Andrew Smith

for the degree of Doctor of Philosophy

of the

University of Bath

Department of Mathematical Sciences

June 2021

COPYRIGHT NOTICE

DECLARATION OF ANY PREVIOUS SUBMISSION OF THE WORK

The material presented here for examination for the award of a higher degree by research has not been incorporated into a submission for another degree.

Signature of Author .....

Cameron Andrew Smith

DECLARATION OF AUTHORSHIP

I am the author of this thesis, and the work described therein was carried out by myself personally in collaboration with my supervisors.

Chapter 2 is reproduced from a *Journal of the Royal Society: Interface* article.

Chapter 3 is reproduced from a *Royal Society: Open Science* article.

Chapter 4 is reproduced from a *Physical Review E* article.

Chapter 5 is reproduced from a prepared and revised manuscript.

Chapter 6 is reproduced from an article submitted to *Journal of the Royal Society: Interface*.

Signature of Author .....

Cameron Andrew Smith

Abstract

Reaction-diffusion systems are of importance in both a biological and physical context, across a multitude of length scales. From the movement and interaction of calcium ions in an intracellular environment to the spread of a contagious disease through a population, reaction-diffusion systems are flexible and can provide, at least to a first approximation, a modelling framework for the evaluation of real-world problems. Mathematically, there are several ways in which we can model reaction-diffusion systems, three of which form the focus of this thesis. At the coarsest scale lie macroscopic models such as partial differential equations (PDEs), which contain no stochasticity but for whose solution there exists a wealth of analytical and numerical techniques. Whilst they can be relatively quick to simulate, they can, however, be inaccurate if complex interactions are present in the system. At a finer level of representation, we have the mesoscale, represented by an on-lattice position jump process, coupled with interaction rules, typically simulated using the Gillespie algorithm or its variants. This method allows for stochastic fluctuations but can be prohibitively slow if there are many particles present. At the finest level we have the microscale, where individual particle locations are tracked and used for the purposes of interaction. This is our most accurate representation, but it is also typically the slowest of the three.

Hybrid methods combine these different representations in order to exploit the advantages, whilst limiting the disadvantages of using each one individually. In particular, this thesis is concerned with so-called "spatially coupled" hybrid methods — those in which the spatial domain is split into two or more regions within which different modelling paradigms are employed, the regions interacting through either an interface or overlap region. Such methods are important when the system under consideration has large spatial variation in particle numbers, or when a particular region of the spatial domain requires more detail. In this thesis, we develop four new hybrid methods, with one on a static domain and the three on growing domains. We also look at developing modelling methods for some of the individual paradigms, focussing on forming equivalence frameworks between different modelling regimes.

ACKNOWLEDGEMENTS

Ah, the acknowledgements. The one part of the thesis that I can truly be myself, and for me one of the most important. It is also the only part nobody else has proofread. So enjoy the next page or two of my unique blend of grammar as I (cliché alert!) thank everyone without whom this wouldn't be possible.

Firstly, to my supervisor Kit. Thank you. Thank you for leading me through the world of research. Your infectious enthusiasm for maths and communication have been a constant source of inspiration to me. I will always be grateful to you for giving me the opportunity to work with you and your group. It really has been a thoroughly enjoyable experience, in spite of code never working! Speaking of the group, a massive thank you has to go to Enrico and Jennifer for the support from the beginning, and recently to Josh and Shahzeb who have provided interesting discussion, and who I wish the best for the rest of their PhDs and beyond.

Next, I need to thank my family, not least because without you, I literally wouldn't be here! Your love and support, not just for the nearly nine years I have been in Bath, but for the 27 years of my existence, has been invaluable and I could not have been here and achieved what I have without you. Mum, Dad and Steve. I know I'm not the best at keeping in touch and coming back to visit. Thank you for keeping me sane, and being there whenever I have needed you. A mention to Lisa too, for keeping Dad sane over the last few years, I will always be grateful. To Alex. I genuinely couldn't haves asked for a better brother. And no, I'm not just saying that because I "have to", but because I mean it, even if your love for Liverpool is bordering on irritating most of the time! I am proud of the man that you have become. And Simonè, I know you have been doing this for a while, but thanks for being his rock. Being in Bath is a lot easier knowing that he has you. Finally, to Granny, Grandad and Auntie Jayne, thank you for the laughs and love through the god and the bad. I can finally say that I have put the brains I got from you, Auntie Jayne, to got use! Jason, I haven't seen my pocket sized Aunt so happy, and that is largely down to you, so thank you. I love you all and hope I have done you proud.

Now for my dear friends, Hannah and David. Thank you for being there, and for being at the end of the phone whenever things went wrong or when I just needed cheering up. The long days and nights chatting about everything and anything, putting the

world to rights (and forgetting to make food, sorry about that!) have kept me going, probably more than you realise. You make me proud thinking of where you are and what you have both achieved, and will continue to achieve, particularly considering you have been through more than a lot of people will have to deal with over their lifetimes. And also a massive thank you for involving me in your most important day. It was one of the biggest honours and highlights of my life to have played such a role in your big day, and to have not fallen and tripped you over while a camera was watching was the icing on the cake, even if it would have made us £250 on You've Been Framed!

To Aoibheann, who has had the unenviable task of living with me for the majority of my PhD. We have had ceiling leaks, a shower that couldn't decide what temperature it wanted to be and a flat that fluctuates between Sahara Desert and Antarctic with very little in between. But it really has been a blast. Thank you for the takeaway nights, the cakes and brownies, chats that go on far longer than they probably should, and reminding me that not everything in Bath is about maths and the PhD. Particularly the last year, where we have been stuck at home together, you have helped make everything that little bit more bearable.

Susie, Paul and Andreas, Jess, Helena, Lindsay and everyone else connected with SAMBa, and also Karsten, Paul (again!) and all those in the department, I extend a massive thank you to you. Not just for giving me a chance here, but for your continued support, even when complaints may have been irritating or trivial for you. I have really enjoyed taking part in or organising the various events that take place, and they have helped make me a better mathematician and person. I wish you all the greatest success in the future, and I am sure SAMBa and the department as a whole will go from strength to strength with so many good people at the helm.

I couldn't get through this without acknowledging the vast number of friends that I have made along the way and groups that I have been a part of. To my cohort, we went through some trying months, but came out the other side. It hasn't always been easy, but I am grateful to have been through it with you. Although I'm not sure I'm too grateful for the Camzabeth face morph that happened! A special mention to my buddy for the final stretch, Hayley, who has been there even when things haven't been great. To all the awesome people who I have shared (procrastination) breaks with in the office, to Abby and Alice for the numerous hot chocolates in the Edge and to Katie and Jeremy for just general tomfoolery that never ceased to make me laugh. There are far too many of you to mention by name, but know that you have all played a massive role in my enjoyment here. To Jack and the two Wills for providing many a walk and

drink in the pub, when they were a thing, and of course the computing support! To anyone who came along to DitP, either in person or online (and in particular to my predecessor Will, and successor Katie), thank you for suffering through my attempts at humorous emails and quizzes. A thank you also needs to be extended to those who have taken part in maths choir, and again to the organisers of that (Ben, Elizabeth, Matt and Zoë) for providing a distraction from everything going on, whilst reigniting one of my hobbies from years ago. To the quiz team, Matt, Ben, Elizabeth, John, the two Toms and Jess, as well as those that came occasionally. Thanks for having me on the time despite my fairly narrow knowledge of sport and flags, but at least I could provide wine with paper aeroplane competitions! And finally to everyone I have played football with through the years, either casually or as part of the Staff and Postgraduate Football League. I have thoroughly enjoyed playing with you all. Particular thanks must go to Rog, Matt and Jacob for helping to run and keep things going.

And finally, I have a few final acknowledgements to make who have made a vital contribution to the last few years. Firstly, thank you to Pepsi for providing me with all the caffeine I could possibly need to work efficiently and sleep as well as an insomniac with someone playing the drums in their room. Although maybe they should be thanking me as I think I have single handedly kept them in business. Secondly, to Spotify for providing the soundtrack to my thesis, which really has stopped me from descending into madness (I think...). And finally to Moles. If you know, you know.

I know I will have forgotten some people in this fairly large ramble, and if that is the case, then I am truly sorry. However, just because my memory is fairly awful, it doesn't mean I don't appreciate what you have done, no matter how small. Thank you. And if you have made it this far down, then thank you for taking the time to read these acknowledgements.

Contents

Chapter 1

Introduction

Reaction-diffusion systems are an important class of models used extensively across the biological and physical sciences. Particles may interact with one another and/or the boundary, and are able to diffuse throughout the spatial domain. These mechanisms are important for the formation of patterns in semi-arid landscapes (Sherratt, 2005), the movement of travelling waves (Moro, 2004), the spread and containment of epidemics (Volpert and Petrovskii, 2009) and intracellular dynamics (Andasari et al., 2012; Khan et al., 2011; ZhuGe et al., 2000) amongst many other examples. There is also significant interest in extending reaction-diffusion systems onto growing domains. Applications for such systems would be domains which grow and shrink, such as neural crest cell migration during embryogenesis (McLennan et al., 2012) and wound healing (Greenhalgh, 1998; Grinnell et al., 1999). One of the most common uses of reaction-diffusion systems, both on static and growing domains, is the study of the formation of patterns. diffusion-driven instability was first conceptualised in Alan Turing's seminal paper "The chemical basis of morphogenesis" (Turing, 1952). Turing's proposed mechanism involved investigating a system of reactions at a stable equilibrium that is driven to instability under the addition of diffusion, causing patterns to form. Recently, domain growth has been shown to be an important component of pattern formation for reaction-diffusion systems (Crampin et al., 1999; Woolley et al., 2011).

Each of these examples demonstrate that it is important to have simulation techniques that are both accurate and efficient in order to simulate reaction-diffusion systems on static and growing domains. There are multiple ways in which we are able to model reaction-diffusion systems, of which we focus on three within this thesis, which we refer to as the macroscale, the mesoscale and the microscale.

At the coarsest scale we have the macroscale. In general, we model at this scale using partial differential equations (PDEs), however, stochastic partial differential equations are becoming more prevalent (Alexander et al., 2002, 2005). PDEs are generally fast to simulate with a wealth of different techniques (finite-difference, finite-volume and finite-element methods (Smith, 1985)), and there are also many analytical techniques that can be utilised, such as steady state analysis, linear stability analysis and the ability to, in some cases, find full analytical solutions. PDEs require a large number of particles in order to be valid. With small particle numbers, stochastic fluctuations are larger in comparison to the system size, and as a result they cannot be ignored.

We focus on finite-difference methods in this thesis, which are a class of methods for approximating the solution of differential equations. We discretise the spatial and temporal domains and describe the solution of the differential equations at those discretised points. Derivatives are approximated as differences on this finite grid (which, for example, are derived by using Taylor's theorem). These differences then yield a matrix-vector system which can be solved in order to evolve the system. Examples of finite-difference schemes applicable to the diffusion dominated equations we consider in this these are the forward Euler method, the Crank-Nicolson method and, more generally, the θ-method. Care needs to be taken when discretising the spatial and temporal domains. Many finite-difference schemes have conditions for the stability of the method. For example, to maintain stability for the forward Euler method we require that the diffusion coefficient multiplied by the time-step divided by the spatial step squared must be less than a half.

In some limited cases, reaction-diffusion PDEs can be solved analytically, even with uniform domain growth. Simpson (2015) describe a method by which reaction-diffusion equations on uniformly growing domains can be solved analytically by utilising a change of variables in both space and time. Firstly, the authors change from the Eulerian coordinates in space to a reference domain, and then the temporal variable is rescaled in such a way that all explicit time-varying terms are moved onto the linear reaction terms. In this way, the usual separation of variables can be used to solve this altered PDE, and then coordinates can be changed back.

The middle scale is the mesoscale, in which we utilise Markovian-based update rules. The spatial domain is split into compartments within which particles reside. Particles may jump between adjacent compartments in order to mimic diffusion, and may interact with others in their own compartment. The system is updated using the Gillespie algorithm (Gillespie, 1977) or any of its variants (for example, the modified next

reaction method (MNRM) which we utilise in Chapters 4 and 6). The method is relatively fast when there are small numbers of particles, but becomes prohibitively slow with larger numbers. However, the mesoscale incorporates stochastic information on particle numbers, which the PDE is unable to do. Moreover, while we do model particle numbers at this scale, their exact positions are lost and instead replaced with the spatial extent of the compartment in which they reside. We also lose particle identity, meaning that once a particle completes an action such as jumping into an adjacent compartment, we no longer know which of the particles in the receiving compartment it is.

The Gillespie algorithm was initially developed to simulate exact solution trajectories for the chemical master equation (CME), which describes the probability distribution of the number of particles of different species in a volume undergoing a number of chemical reactions. The basic premise is to calculate the time until the next reaction takes place, and then to decide which reaction occurs proportional to their rates. Since the original algorithm was popularised in 1977 (Gillespie, 1977), several variations have been produced which are both exact and inexact. Methods such as the next reaction method (Gibson and Bruck, 2000), the modified next reaction method (Anderson, 2007) and the next subvolume method (Elf and Ehrenberg, 2004) are all variations which are designed to increase the efficiency or to relax some of the assumptions (such as allowing explicit time-dependent propensity functions) of the original method in particular scenarios. Inexact methods sacrifice some of the accuracy of the exact method for increases in efficiency. One such example, and one of the more commonly employed, is τ-leaping (Gillespie, 2001), which takes large temporal jumps and then approximates the number of reactions that have taken place in that time gap. Other inexact methods include binomial leaping (Tian and Burrage, 2004) and R-leaping (Auger et al., 2006).

Regardless of the numerical scheme that is employed, we need to be able to connect the mesoscale to the macroscale. We refer to this as an "equivalence framework" throughout the thesis. The idea of the framework is to relate the parameters for the different processes (diffusion, growth, reactions) at the three scales considered. To connect the mesoscale to the macroscale, we calculate the reaction-diffusion master equation (RDME) for the system in question. We then calculate the mean number of particles in a given compartment and take the diffusive limit in order to obtain the mean-field equations. If we have second- or higher-order reactions, we employ moment closure in order to obtain a single PDE for the mean of the system (see, for example, Section 4.4.1 of Chapter 3 for an example). Using this process, we are able to link the rate of jumping between compartments at the mesoscale to the Fickian diffusion

coefficient in the PDE, and the rate of stretching to the exponential growth rate of the domain (when this is applicable). The calculation for these can be found in the supplementary material of the paper in Chapter 6 (Section S.2.3).

At the finest scale is the microscale. In the context of this thesis, microscale refers to an off-lattice method in which we track each particle's location. Particle positions are updated according to a stochastic differential equation (SDE). The probability density of the particle's position can be described by a PDE which is a scaled version of the density for non-interacting particles. To simulate this SDE throughout this thesis, we employ the Euler-Maruyama method, which is a finite-difference method and an extension of the forward Euler method for SDEs rather than ODEs. Further, for second-order reactions to be completed, there are many popular methods in the literature. One such method is the λ-ρ method (Erban and Chapman, 2009). In this method, pairwise distances between particles are required. Appropriate pairs of particles within a distance ρ react with a rate λ, which depends on the kinetic rate of the reaction. Reactions at zeroth- or first-order are executed by adding particles or allowing each particle to react respectively. The microscale is the least efficient of the three methods, but is also able to capture the most detail.

The microscale links to the macroscale through a combination of the Fokker-Planck equation (FPE) and the set of ODEs formed under the mean-field assumption (which gives rise to the same ODEs as if the law of mass action is employed). In the case of no particle interaction, with zero-flux boundary conditions (as is considered in this thesis), the Fokker-Planck equation should be multiplied by the (fixed) number of particles in the system in order to match the diffusion PDE. Once interactions are added, we take the scaled FPE as before and add the mean-field ODEs to it in order to obtain a reaction-diffusion PDE. The equivalence framework for the diffusive and growth processes are found by choosing the drift and diffusion functions in the SDE appropriately so that the scaled FPE matches the diffusion equation on a growing domain (Risken, 1996). For reactions of second order, given a kinetic reaction rate and a choice of reaction radius, we are able to find the probability of interaction within that radius using the λ-ρ method (Erban and Chapman, 2009). This then matches with the mean-field PDE, for which the appropriate term is the kinetic rate multiplied by the densities of the two interacting agents (see Section 4.4.1 of the paper in Chapter 3).

Each of these methods (macroscale, mesoscale, microscale) is suitable for different numbers of particles, and have different levels of detail and speed. Spatially extended hybrid methods are able to couple two or more of these modelling paradigms together in

order to negate the complimentary disadvantages whilst exploiting the complimentary strengths of each method. The spatial domain is split into distinct or overlapping subdomains within which different modelling methods are employed. Separating each of the subdomains is an interface, over which there are rules which specify how to convert mass between the two regions. There are many examples of spatially extended hybrid methods, as well as hybrid methods of other types (see Chapter 2 for more information).

Aside from the benefits in accuracy and efficiency that are described above, hybrid methods may be able to help answer some biological questions that may have been previously infeasible. As an example, consider the release of calcium in the intracellular environment via the endoplasmic reticulum. Not only are local calcium "puffs" formed, but there are also global waves which propagate throughout the cell. This example demonstrates two behaviours of interest at different scales: the localised puffs which occur close to the boundary and the global waves which are caused by these local puffs throughout the cell. In order to capture the localised calcium puffs, a method such as the Brownian-based dynamics would be required. However, this would be expensive to employ further away from the ion channels, particularly when it would not be needed. Dobramysl et al. (2015) simulate this scenario using a meso-micro hybrid method, allowing for both the local and global behaviours to be captured without sacrificing too much efficiency.

Hybrid methods have the potential to yield an increase in efficiency compared to that of the finest scale modelling paradigm contained within the method. They are also, by design, relatively simple to implement if the user has knowledge of how to simulate each of the individual elements of the hybrid method. Consequently, they have the advantage of giving their gains in efficiency with little extra effort when coding up the numerical scheme.

Throughout this thesis, we look to match our hybrid methods at the mean-field level. The primary reason for this is because we are looking to develop the algorithms and ensure that they pass the most basic of tests. However, there are more questions that can be asked of these hybrid methods. Firstly, one could consider the variance of the system, which we briefly look at in Chapter 3. Other considerations that may be of interest to those who use hybrid methods may be first time of arrival at a certain point in space, the switching time behaviour between two stable steady states or the behaviour of higher order moments. As of now, very little work has been done beyond investigating mean-field and variance properties of hybrid methods. This is something

we reflect on in the discussion chapter (Chapter 7).

The work within my PhD has been centred on developing spatially extended hybrid methods on growing domains, and has led to the creation of a further hybrid method on a static domain, together with a couple of papers that investigate equivalence of various representations on growing domains. As a summary, we begin with Chapter 2 which does not contain new work, but is instead a review and guide for spatially extended hybrid methods. In Chapter 3 we develop a new macroscopic-to-microscopic hybrid method called the auxiliary region method (ARM) (Smith and Yates, 2018) which we later extend onto a uniformly growing domain (see Chapter 6). This method was created because there was no suitable hybrid method that reliably coupled macroscopic and microscopic methods. We found the existing method was unreliable in many parameter regime and therefore decided to create a new one. Following this, we have two papers which establish equivalence frameworks. In Chapter 4 we develop the stretching method (Smith et al., 2019), a mesoscopic method for domain growth which does not have an artificial build up of particles at the boundaries, as we discovered in certain parameter regimes for a previous method. In Chapter 5 we investigate the general Robin boundary conditions for a PDE, and how to achieve the same condition for the two stochastic equivalents on a growing domain. This paper is required for a couple of the test problems for the growing hybrid methods. Finally, in Chapter 6, we bring all of these developments together to create three growing hybrid methods, one for each of type (macroscopic-to-mesoscopic, mesoscopic-to-microscopic and macroscopic-to-microscopic) which extend the pseudo-compartment method (PCM) (Yates and Flegg, 2015), the ghost cell method (GCM) (Flegg et al., 2015) and the auxiliary region method (ARM) (Smith and Yates, 2018).

While the work is not included within this thesis, there is one further novel hybrid method whose development, and subsequent publication, I have been involved with. This method is called the blending method (Yates et al., 2020). This work, published in the *Journal of the Royal Society: Interface* contains both a macroscale-to-mesoscale and a mesoscale-to-microscale method, each using a similar coupling idea. The two methods that are to be employed are coupled through the use of a blending region, within which both representations are valid. Through the use of complementary "blending functions", the implementation of diffusion is passed from one modelling paradigm to the other as the blending region is traversed from one side to the other. These blending functions are complimentary in the sense that they sum to give the overall diffusion coefficient for every spatial value, so that the method is equivalent to the usual diffusion equation.

1.1 Summary of original papers

Chapter 3 — The auxiliary region method: a hybrid method for coupling PDE- and Brownian-based dynamics for reaction-diffusion systems

In this chapter, we develop a novel spatially extended hybrid method that couples a macroscopic PDE representation of reaction-diffusion systems to an individual-based particle description, separated by an interface, called the auxiliary region method (ARM). We created this method because a previous interfacial method, presented by Franz et al. (2013), is not suitable due to the impact of parameter changes on its accuracy. The ARM relies on an intermediate scale, the mesoscale, in order to allow particles to pass across the interface and change their representation. We demonstrate that the method is accurate whilst also being robust to parameter changes, something which the method of Franz et al. (2013) fails to achieve.

Chapter 4 — Unbiased on-lattice domain growth

We present a method for modelling a reaction-diffusion system on a growing domain using a Markovian, on-lattice approach, which we commonly refer to as the mesoscale. The approach relies on the assumption that mass should spread out uniformly when the domain is being stretched. We show that the method gives mean particle densities that are equivalent to the diffusion PDE on a growing domain in the small box-size limit, and that our method is able to accurately mimic this PDE. We finally show that the method is applicable to low diffusion regimes, where the domain grows at a much faster rate than the particles diffuse. This is a behaviour that a previous method (Baker et al., 2010) is unable to replicate. Instead, that method causes an artificial build up of particles at the boundaries in such growth-dominated, low-diffusion regimes, a problem which our method is able to avoid.

Chapter 5 — Robin boundary conditions on growing domains: equating PDE to stochastic representations

In this chapter, we develop an equivalence framework for a general Robin boundary condition on a growing domain, defined via a partial differential equation. The main motivation for the work of this chapter is to be able to employ a flexible boundary condition for a range of test problems, particularly when we investigate the accuracy of our growing hybrid methods. We equate the method to two different stochastic representation: the first being the Markovian, on-lattice method, while the second is the off-lattice, Brownian-based dynamics. We demonstrate that the equivalence is

accurate through the use of three test problems.

Chapter 6 — Incorporating domain growth into hybrid methods for reaction-diffusion systems

In this chapter, we bring together all of the work of the previous chapters in order to extend three spatially extended hybrid methods onto a uniformly growing domain. Each of the three methods are of a particular type, namely compartment coupled, which means that the mesoscale is required in order to allow particles to jump over the interface. The three algorithms cover the complete spectrum of spatially extended hybrid methods (macro-to-meso, meso-to-micro and macro-to-micro) and are shown to be accurate over three test problems designed to assess the ability of the hybrid methods to perform under a range of scenarios.

1.2 A note on the alternative thesis format

The alternative thesis format allows for chapters to be made up of papers that have been published, submitted for peer-review or are in preparation. Each paper also contains a commentary in the form of an introduction and conclusion, which sets it in the wider context of the thesis.

Chapter 2 contains a paper published in the *Journal of the Royal Society: Interface*. Chapter 3 contains a paper published in *Royal Society: Open Science*. Chapter 4 contains a paper published in *Physical Review E*. Chapters 5 and 6 contain papers which have been completed recently. The paper contained in Chapter 5 is currently in preparation and presented in its near final draft form, while the paper in Chapter 6 has been submitted to the *Journal of the Royal Society: Interface*. All papers, regardless of their publication status, will be presented in manuscript format, and the page number at the bottom of the page is the page number for the entire thesis. I am the first author of all papers presented within this thesis.

Chapter 2

Spatially extended hybrid methods: a review

This chapter contains a review paper published in *Journal of the Royal Society: Interface* in 2018, authored by myself and Kit Yates. The paper serves two main purposes. The first is to collate the literature on spatially extended hybrid methods (those which split space into distinct subdomains on which different modelling paradigms are utilised) so that if anybody wishes to use one in their work, they have a reference document which contains a description of several of each type (macroscopic-to-mesoscopic, mesoscopic-to-microscopic and macroscopic-to-microscopic) of spatially extended hybrid method. The second is to provide illustrative examples of each type, together with algorithms for their implementation and some sample results, coupled with the code, so that there are concrete case studies that are able to be utilised. In this respect, the paper can be seen as both a reference and user guide for spatially extended hybrid methods.

This paper lays the foundations for the rest of the thesis, putting the papers that will come in the following chapters into context (as of 2018). Moreover, the three canonical methods we outline here appear several more times throughout this thesis. In particular, the macroscopic-to-microscopic method employed in Section 5 of this review paper is the auxiliary region method (Smith and Yates, 2018), which will be explained in its entirety in Chapter 3. All three illustrative examples will later be extended onto uniformly growing domains in Chapter 6.

After the paper, in the discussion section of this chapter, we review some of the spatially

extended hybrid methods that have been developed since the publication of our review paper.

2.1 Outline of paper

The paper in this chapter begins with an introduction to spatially extended hybrid methods, and a list of possible software packages for each of the individual modelling paradigms or some hybrid methods in Section 1. Section 2 contains a mostly qualitative description of the macro, meso and microscales, and how they could be implemented. This section also contains a brief explanation on how these different modelling paradigms could be considered equivalent. Section 3 contains a review of hybrid methods which couple the macroscale to the mesoscale, including an in-depth look at the pseudo-compartment method (Yates and Flegg, 2015), including a description, algorithm for its implementation and some numerical results. Section 4 contains the same for the mesoscopic-to-microscopic methods, with the illustrative example being the ghost cell method (Flegg et al., 2015). Section 5 is once again the same as Sections 3 and 4, this time investigating macroscopic-to-microscopic methods, and a closer look at the auxiliary region method (Smith and Yates, 2018). Section 6 contains a look at some other hybrid methods that do not fit into the categories of Sections 3-5. Finally, in Section 7 there is a discussion of hybrid methods and their potential use in the future.

Appendix 6B: Statement of Authorship

This declaration concerns the article entitled:			
Spatially extended hybrid methods: a review			
Publication status (tick one) **Draft manuscript** ☐ **Submitted** ☐ **In review** ☐ **Accepted** ☐ **Published** ●			
Publication details (reference)	Journal - Journal of the Royal Society: Interface, 15 (139), 20170931 Authors – Cameron A. Smith, Christian A. Yates		
Copyright status (tick the appropriate statement) I hold the copyright for this material ● Copyright is retained by the publisher, but I have been given permission to replicate the material here ☐			
Candidate's contribution to the paper (provide details, and also indicate as a percentage)	All calculations have been performed by the author of this thesis (100%). All numerical computations and simulations have been completed by the author of this thesis (100%). All authors contributed equally to the presentation of the content (50%).		
Statement from Candidate	This paper reports on original research I conducted during the period of my Higher Degree by Research candidature.		
Signed		**Date**	04/06/2021

Spatially-extended hybrid methods: a review

Cameron A. Smith[1,*], Christian A. Yates[1,**]
1 Centre for Mathematical Biology, Department of Mathematical Sciences, University of Bath, Claverton Down, Bath, BA2 7AY, United Kingdom
∗ **E-mail: c.smith3@bath.ac.uk**
∗∗ **E-mail: c.yates@bath.ac.uk**

Key index words: hybrid modelling, reaction-diffusion, multiscale, modelling

Abstract

Many biological and physical systems exhibit behaviour at multiple spatial, temporal or population scales. Multiscale processes provide challenges when they are to be simulated using numerical techniques. While coarser methods such as partial differential equations are typically fast to simulate, they lack the individual-level detail that may be required in regions of low concentration or small spatial scale. However, to simulate at such an individual-level throughout a domain and in regions where concentrations are high can be computationally expensive. Spatially-coupled hybrid methods provide a bridge, allowing for multiple representations of the same species in one spatial domain by partitioning space into distinct modelling subdomains. Over the past twenty years, such hybrid methods have risen to prominence, leading to what is now a very active research area across multiple disciplines including chemistry, physics and mathematics.

There are three main motivations for undertaking this review. Firstly, we have collated a large number of spatially-extended hybrid methods and presented them in a single coherent document, while comparing and contrasting them, so that anyone with a need for a multi-scale hybrid method will be able to find the most appropriate one for their need. Secondly, we have provided canonical examples with algorithms and accompanying code, serving to demonstrate how these types of methods work in practice. Finally, we have presented papers that employ these methods on real biological and physical problems, demonstrating their utility. We also consider some open research questions in the area of hybrid method development and the future directions for the field.

1 Introduction

The requirement for multi-scale models arises naturally from many biological and physical scenarios due to their inherent complexity. However, modelling such systems is often difficult using a single modelling paradigm. This is due to the fine balance between acquiring results in a timely manner (efficiency) and obtaining results that are consistent with the experimentally derived knowledge or physical laws (accuracy). One such example is modelling the release of calcium from the endoplasmic reticulum, and its subsequent movement throughout the cell (Dobramysl et al., 2015; Flegg et al., 2013). Calcium ions leave the endoplasmic reticulum through ion channels which open or close depending on whether other calcium ions have bound to receptors. The behaviour of calcium ions close to the receptors can only be simulated using an individual-based method, as we require the knowledge of every particles' locations. However, when the channel opens, a large number of particles enter the cytoplasm of the cell. Keeping track of all of these particles is computationally costly, leading to limitations on the time-scales which can feasibly be simulated using the fine-grained model alone.

This review will focus on four modelling scales. The first of these is the macroscopic scale. This encompasses all models in which we make the assumption of large copy numbers within the system, such as partial differential equations (PDEs) or stochastic partial differential equations (SPDEs). In most cases, these continuum models can be simulated extremely efficiently, but they are generally invalid for low numbers of particles.

At the next finest scale is the mesoscopic scale. Typically, models at this scale employ stochastic methods in which particles are compartmentalised into small subregions of the domain, within which they are assumed to be well-mixed. Particles can transfer between compartments, and interact with other particles within their own compartment, according to a Markov chain. Models at the mesoscale can be fast to simulate with small copy numbers, but when these become large, the method can become prohibitively slow.

On an even finer scale, we have microscopic models. These simulate the trajectory of each particle in the system (typically using a fixed time-step algorithm), requiring their locations to be updated at each time-step. Examples of individual-based microsopic models include Brownian dynamics (Andrews and Bray, 2004; Smoluchowski, 1917) or Langevin dynamics (Langevin, 1908). These methods can be very computationally intensive. For example, for a system of N particles undergoing Brownian dynamics, at each time-step, we are required to generate δN Gaussian random variables (where δ is the dimension of the system) in order to update the positions of the particles. In addition, if pairwise interactions are necessary, the calculation of N^2 pairwise distances is required. For large N this can be the limiting step in the method. While costly, microscopic individual-based dynamics do allow for a high level of modelling accuracy, which is often required.

On the very finest scale are molecular dynamics (Dürr et al., 1981; Holley, 1971). In a typical molecular dynamics simulation, a large number or particles ($\sim 10^{10}$) with attributes of mass, momentum and volume-exclusion are simulated with an extremely small time-step (typically around 10^{-15} s). The position and velocity of all particles are updated according to deterministic equations specified by conservation of mass, momentum and energy. Because of the very small time-scales and enormous number of molecules, these simulations are extremely computationally expensive. However, they are necessary in order to accurately resolve the fine-level detail that is crucial for many sub-cellular processes including, for example, protein-protein interactions (Plattner et al., 2017).

The term 'hybrid method' has come to mean many different things in the modelling literature. Typically, it refers to computational methods which represent phenomena using more than one modelling paradigm. Usually, the reason for multiple modelling paradigms is a significant separation in scale. This separation may be in time scales (Cao et al., 2005; Hellander et al., 2012; Klann et al., 2012), in species copy number (Anderson, 2005; Franz et al., 2013b) or in spatial scales (Dobramysl et al., 2015). By coupling an expensive, but accurate 'fine-scale' model to a cheaper, but less accurate, 'coarse-scale' model, hybrid methods allow for the significant acceleration of simulations that would be computationally expensive if the fine-level model were used for all components of the system or inaccurate if the coarse-level model were employed ubiquitously.

There are range of hybrid methods that have been developed to model well-mixed systems (Bentele and Eils, 2004; Bobashev et al., 2007; Burrage et al., 2004; Duncan et al., 2016; Hellander and Lötstedt, 2007; Hepp et al., 2015; Kiehl et al., 2004; Salis and Kaznessis, 2005). These methods typically exploit a separation of time-scales in which fast reactions or abundant species are modelled using a coarse description and slow reactions or scarcer species are modelled using a more accurate finer description.

However, if the spatial extent of a system is important (when modelling pattern formation, travelling waves and chemotaxis (Murray, 2003), for example) then there are an even broader range of *spatially-extended* hybrid methods which employ different modelling paradigms at different scales in order to complement the strengths and negate the weaknesses of each.

If individual species are present in very different concentrations throughout the domain (for example, in the context of chemotaxis, cells are present in low numbers whilst the chemical signalling molecules with which they interact are present in high copy numbers (Dallon and Othmer, 1997; Erban, 2004; Franz and Erban, 2011; Guo et al., 2008; Xue and Othmer, 2009)), distinct modelling paradigms can be used to represent each species in the same simulation. The particular representation will depend on the abundance of each species (Alarcón et al., 2003; Anderson, 2005; Anderson and Chaplain, 1998; Dallon and Othmer, 1997; Dormann and Deutsch, 2002; Franz and Erban, 2011; Franz et al., 2013b; Gerlee and

Anderson, 2007; Jackson et al., 2006; Jeon et al., 2010; Jeschke and Uhrmacher, 2008; Landsberg and Waring, 1997; Osborne et al., 2010; Patel et al., 2001; Ribba et al., 2004; Smallbone et al., 2007; Wylie et al., 2006). Other types of spatial hybrid method partition the physical processes (for example reactions and diffusion) to be simulated according to their relative speeds, using a technique known as operator splitting (Hellander et al., 2012; Klann et al., 2012), simulating faster processes using relatively cheap methods and slower processes using more accurate but more expensive representations.

For the purposes of this review, we will largely focus on methods in which distinct modelling paradigms are used in different regions of space in order to represent the same physical quantity. The models in these distinct regions of space are typically coupled together though an interface or overlap region. Spatially-coupled hybrid methods, of the sort we cover in this review, rely on the assumption that different regions of the spatial domain can be accurately represented using modelling paradigms at different scales (Erban, 2014; Flegg et al., 2012, 2015; Smith and Yates, 2017; Yates and Flegg, 2015). The motivation for these methods will typically be either a separation in the scale of species copy numbers in distinct regions of the domain or a requirement for a detailed model on small spatial scales.

Widely differing species copy numbers in distinct regions of the domain allow coarse models to cheaply capture the dynamics in regions in which copy numbers are high whilst a fine model captures the details of low copy number populations with the required accuracy. Typically these methods would be used for phenomena which are multiscale in copy number, such as travelling wave problems (Moro, 2004; Robinson et al., 2014). Behind the wave we have large copy numbers meaning that a coarse description can be used. At the wave front and further ahead, however, stochastic variation will play a more important role in determining the correct dynamics. Consequently, a fine description is required in these regions.

Alternatively, even if there is no significant difference in copy numbers throughout the domain, there may be a small region of space which requires fine-level modelling locally, but which can tolerate coarser modelling further away in regions which are not sensitive to the individual dynamics. Typically, these methods are used to represent phenomena in which boundary effects are important (Dobramysl et al., 2015).

We will refer to these methods (whatever the underlying motivating dynamics) as *spatially-coupled hybrid methods*. Although we will largely focus on these spatially-coupled hybrid methods in this review, we will also touch upon other the hybrid methods which accelerate spatially-extended stochastic simulations where appropriate.

While a full description of each is beyond the scope of this review, we nevertheless reference numerous software packages designed to simulate systems at each of the four spatial scales described above (typically individually, but occasionally incorporating hybrid dynamics), which are summarised in Table 1. For more information on any of these software packages, we refer the reader to the appropriate reference, which is given in the final column of the table.

In this paper, we review some of the vast array of hybrid methods present in the literature. In Section 2, we introduce the four most popular modelling paradigms for reaction-diffusion systems at different scales. In Sections 3, 4 and 5 we review the three main forms of spatially-coupled hybrid method. Each of these sections will begin with an in-depth review of an illustrative example, including pseudocode for its implementation, before we summarise other existing hybrid models of that type. Following these, in Section 6, several other types of hybrid method will be reviewed, before we conclude in Section 7.

2 Modelling paradigms

Within this section, we will describe modelling paradigms that are coupled most often in order to create hybrid methods. In Section 2.1 we describe a general PDE for reaction-diffusion systems with a single species. Section 2.2 contains an outline of compartment-based models, while in Section 2.3 we investigate individual-based dynamics. In Section 2.4, we briefly introduce molecular dynamics, and finally in Section 2.5 we indicate how each of these modelling methods can, in some sense, be demonstrated to be equivalent

Software Package	**Uses**	**Types**	**Reference**
Copasi	Next reaction method, Hybrid methods	Meso, Macro-meso	Hoops et al. (2006)
E-Cell	Direct method, Next reaction method, τ-leaping	Meso	Tomita et al. (1999)
Lattice Microbes	Direct method, Next reaction method	Meso	Roberts et al. (2013)
MCell	Spatial stochastic simulation	Meso, Micro	Stiles and Bartol (2001)
Smoldyn	Spatial stochastic simulation	Meso-micro	Andrews and Bray (2004)
STEPS	Direct method	Meso	Wils and De Schutter (2009)
StochKit	Direct method, Optimised direct method, τ-leaping, stochastic simulation algorithm	Meso	Li et al. (2008)
(py)URDME	Next subvolume method	Meso	Drawert et al. (2012)

Table 1. Summary of software implementations and the scales which they can be used to model. The table contains only packages that have been updated since 2013. All have been downloaded to test that the links still work. Adapted from Pahle (2009).

representations of reaction-diffusion.

2.1 Macroscopic models

Macroscopic models encompass ordinary differential equations (ODEs) and stochastic differential equations (SDEs) in a well-mixed context, and partial differential equations (PDEs) and stochastic partial differential equations (SPDEs) in a spatially-extended context. PDEs, with which we shall primarily be concerned in this review, are used to model the mean-field behaviour of particles, provided they are at a sufficiently high concentration, whilst SPDEs fulfil the same purpose but with the additional ability to incorporate stochasticity in particle numbers/concentrations. These macroscopic methods can be simulated efficiently, but can fail to correctly capture the appropriate behaviour at low copy numbers, in which the combination of stochastic fluctuations, small particle numbers and potentially non-linear reactions can cause significant discrepancies between the true individual-based dynamics and those of their continuum counterparts.

The methods discussed in this review which employ (S)PDEs are all designed to simulate reaction-diffusion systems, mostly comprising a single species. The PDE for the concentration of a single species, $c(\boldsymbol{x}, t)$, at position $\boldsymbol{x}$ and time t has the general form:

$$\frac{\partial c}{\partial t}(\boldsymbol{x}, t) = D\nabla^2 c(\boldsymbol{x}, t) + \mathcal{R}(c(\boldsymbol{x}, t), \boldsymbol{x}, t), \quad \boldsymbol{x} \in \mathbb{R}^\delta, \quad t \in [0, T] \tag{1}$$

with appropriate boundary and initial conditions. Here D is the diffusion coefficient, $\mathcal{R}$ is a function representing the reactions and δ is the dimension of the space which we are modelling. These systems of PDEs are, in general, very difficult or impossible to solve analytically, especially when second- or higher-order reactions are involved making the reaction function $\mathcal{R}$ non-linear. Typically, however, they can be solved straightforwardly using numerical approximations. One popular family of numerical solution techniques, employed in many of the papers discussed in this review, are finite-difference methods[1] such as the forward Euler or Crank-Nicolson methods. Finite-difference methods discretise the spatial and temporal domains onto a mesh, upon which the PDE solution is approximated. The PDE (1) is converted into a system of difference equations which relate the solution at the next time-step to the solution at previous time-steps. Often, these systems of difference equations may be approximated to first order to form a linear system. There are many efficient techniques for solving such linear systems (see for example (Brenner and Carstensen, 2004; Eymard et al., 2000; Morton and Mayers, 2005; Smith, 1985)), giving a fast method for obtaining a numerical solution of PDE (1).

In this review, in-keeping with the terminology used throughout the reviewed papers, these models will be described as "macroscopic" and, in the deterministic case as "mean-field".

2.2 Compartment-based methods

Compartment-based methods are a coarse-grained stochastic representation. The spatial domain is split into a number of compartments of size h_c, which are assumed to contain uniformly distributed, well-mixed particles. The system can be simulated using either a time-driven or an event-driven algorithm. In both cases, an event is defined as either a diffusive jump, in which a particle jumps from one compartment to a neighbour with rate $d = D/h_c^2$ (here D is the corresponding macroscopic diffusion coefficient) or a reaction, in which particles interact within a compartment according to a specified reaction pathway.

Time-driven algorithms assume a time-step, Δt, that is small enough so that at most one "event" occurs in the time interval $[t, t + \Delta t)$ (Erban et al., 2007). A scaled uniform random number is used to decide whether an event takes place, and if so, which event it is.

Event-driven algorithms are generically known in this context as stochastic simulation algorithms (SSAs). The most commonly used SSA is the Gillespie direct method (Gillespie, 1977), an exact SSA

[1] Note that finite-volume and finite-element methods may work equally well depending on the PDE.

in which each event, represented by a propensity function, has an exponentially distributed waiting time. Consequently, the minimum waiting time of all the events is also exponentially distributed with a rate which is the sum of the rates of the individual reactions. The direct method, thus, simulates an exponential waiting time for the next reaction of *any type* to occur and then the specific reaction to be implemented is chosen with probability proportional to its propensity function. This method is exact in the sense that it simulates the corresponding chemical master equation (CME) exactly. Although this basic method accurately simulates the underlying dynamics, it can be quite slow, and so other, faster methods have been formulated (Cao et al., 2004; Elf and Ehrenberg, 2004; Gibson and Bruck, 2000; Li and Petzold, 2006; McCollum et al., 2006; Yates and Klingbeil, 2013). Additionally if some moderate sacrifices in accuracy are acceptable, several approximate simulation algorithms are available, including τ-leaping and R-leaping (Auger et al., 2006; Gillespie, 2001).

The spatially-extended methods described in this section will be referred to as "compartment-based", "mesoscopic" or "stochastic" (the latter only when coupled with a deterministic model) throughout this report.

2.3 Individual-based modelling

The next set of methods we will consider are individual-based methods. These methods are very computationally intensive for large numbers of particles because they require the storage and maintenance of the positions of potentially large numbers of particles. If second- or higher-order reactions or volume exclusion is to be represented, we need to consider pairwise interactions. The calculation of pairwise distances can also contribute significantly to the cost of these detailed algorithms. In many biologically realistic situations, we may be modelling large numbers of objects at the atomistic scale. In the process of calcium induced calcium release, for example (Dobramysl et al., 2015), there could be tens of thousands of ion positions to keep track of, as well as millions of potential pairwise interactions.

One method of simulating diffusing particles on an individual level is to allow the particles to follow Brownian trajectories, such that:

$$\boldsymbol{y}_i(t + \Delta t) = \boldsymbol{y}_i(t) + \sqrt{2D\Delta t}\ \boldsymbol{\xi}_i, \tag{2}$$

where $\boldsymbol{y}_i(t)$ is the position of particle i at time t and $\boldsymbol{\xi} \sim MVN(\mathbf{0}, I_\delta)$ is a $\delta-$dimensional unit Gaussian random variable. Reactions can then be simulated in a number of different ways. One method, called the λ-ρ model (Erban and Chapman, 2009), uses a reaction radius: if two eligible particles come within a certain distance of one another, ρ, they react with a given rate, λ, according to the appropriate reaction pathway. If this probability is unity and the reaction is certain to occur upon particles reaching the reaction radius, we have the special case of the "Smoluchowski" model (Smoluchowski, 1917). Green's function reaction dynamics are an alternative event-driven microscopic model for simulating reaction-diffusion dynamics (van Zon and ten Wolde, 2005), but since none of the hybrid methods discussed herein employ it, we shall not discuss it further.

We will refer to these methods as "individual-based", "microscopic", "particle-based" or "off-lattice" models in what follows.

2.4 Molecular dynamics

At the very finest scale lies molecular dynamics (Dürr et al., 1981; Holley, 1971). In molecular dynamics simulations, the molecules for the medium in which a particle of interest is moving (air, water etc.) are explicitly modelled rather than implicitly incorporated into the movement dynamics of the focal particle, as is the case with random position jumps of Brownian motion models, for example. For coarse molecular dynamics representations (as opposed to fully atomistic simulations), the particles of the medium can be considered to be identical hard spheres with a given radius and mass and whose velocity and hence

momentum are specified initially, but change dynamically throughout the simulation. Particles interact with each other and in such a way as to conserve mass and momentum.

Although the resulting motion of the large focal particle may appear stochastic, it is in fact calculated deterministically by considering the many interactions with each of the small particles in the surrounding fluid, as well as the larger microscopic particles. Whilst this method of modelling explicitly accounts for the surrounding molecules instead of modelling them as a stochastic force (as in an individual-based method), keeping track of the large number of particles of the medium, their coordinates and their velocities, is computationally intensive.

2.5 Connections between models at different scales

In order to couple models at different scales together, we first need to be satisfied that they are representations of the same phenomena. Here we briefly detail how the different scale models described above can, in some senses, be thought to be equivalent to each other. We direct the interested reader to appropriate sources for full derivations.

Firstly, in order to move from the mesoscale to the macroscale, we take the diffusive limit of a set of equations for the mean number of particles in each compartment, derived directly from the reaction-diffusion master equation (Erban and Chapman, 2009). In the case of second- and higher-order reactions, the mean equations depend on higher order moments (variance etc.). As a result, moment closure is required in order to close the system. The most common moment closure at first order is known as the mean-field moment-closure and the resulting equations are known as the mean-field equations. It should be noted that the mean-field PDEs derived in the case of second- and higher-order reactions, therefore, are not exact descriptions of the mean behaviour of the mesoscale model (Erban et al., 2007). To derive the corresponding macroscale model of diffusion from the microscale model, one can use the Fokker-Plank equation, which describes the evolution of the probability density of a particle moving according to a given SDE (Erban et al., 2007). For example, the Fokker-Planck equation corresponding to non-interacting particles undergoing simple Brownian motion is the canonical diffusion equation. The mesoscopic and microscopic representations can therefore be thought of as equivalent, in some sense, through their connection to the PDE. A rigorous derivation of the connections between the models at microscale and mesoscale is given by Isaacson (2008). Finally, the motion of a large focal particle buffetted by smaller particles of medium as part of a coarse molecular dynamics simulation, has been shown, in the limit that the focal particle's mass becomes large in comparison to the mass of the particles of the medium, to be equivalent to Brownian dynamics (Erban, 2014).

3 Macroscopic-to-mesoscopic models

In this section, we will first introduce the broad concept, and then review specific examples of models which couple macroscopic dynamics to mesoscopic dynamics, which we will refer to as "macro-meso" hybrid methods. We list and describe the macro-meso hybrid methods covered in this section in Table 2. We begin by giving an illustrative example of a macro-meso hybrid method, the pseudo-compartment method (PCM) (Yates and Flegg, 2015) and present pseudocode for its implementation. We then summarise several other existing macro-meso hybrid methods and present schematics (where appropriate) to aid the reader's understanding.

Macro-meso models are used when we want to simulate a region of the domain in which stochastic variation is important but in which the exact locations of every particle are not required, whilst for the remainder of the domain we have sufficiently high copy numbers to employ the associated continuum model. Typical examples to which these hybrid methods have been applied are the simulation of travelling wave phenomena (Harrison and Yates, 2016; Moro, 2004). Behind the wave-front, we have a large number

Paper	Type	System modelled
Yates and Flegg (2015)	Spatially-coupled, non-adaptive, non-overlap	Reaction–diffusion
Moro (2004)	Spatially-coupled, non-adaptive, non-overlap	Reaction–diffusion
Spill et al. (2015)	Spatially-coupled, adaptive, non-overlap	Reaction–diffusion
Schulze et al. (2003)	Spatially-coupled, adaptive, no-overlap	Epitaxial growth
Harrison and Yates (2016)	Spatially-coupled, adaptive, overlap	Reaction–diffusion
Flekkøy et al. (2001)	Spatially-coupled, non-adaptive, overlap	Reaction–diffusion
Rossinelli et al. (2008)	Operator splitting	Reaction–diffusion
Lo et al. (2016)	Operator splitting	Reaction–diffusion
Chiam et al. (2006)	Propensity-based spatial splitting	Reaction–diffusion

Table 2. A summary of the macro-meso hybrid papers that will be covered in this section. The "type" column gives a brief description of the type of coupling used to join the two regimes. "Spatially-coupled" means that the domain is split into two distinct regions within which different paradigms are used. "Adaptive" refers to whether an interface is able to move, while "overlap" indicates if an overlap region is investigated. "Operator splitting" indicates where reaction and diffusion are modelled in different ways, rather than dividing space, and "propensity-based spatial splitting" is where the propensity functions are split based on their value. The "system modelled" column describes the application for which these models can be used. All of the macro-meso hybrid papers present novel methods rather than applications of pre-existing methods to real-world systems.

of particles so that the continuum limit is valid, whilst in front of the wave, fluctuations can play a prominent role in the overall dynamics, including the wave speed.

3.1 Illustrative example of a macro-meso hybrid – the pseudo-compartment method

The first macroscopic-to-mesoscopic example we present is the pseudo-compartment method (PCM) (Yates and Flegg, 2015). We will treat this method as an illustrative example for this section, and as such, will present it in a high level of detail, including a schematic (see Figure 1) and pseudo-code (see Algorithm 1). Note that for all three illustrative examples, we set the dimension of space to be $\delta = 1$ for simplicity.

The authors divide their domain of interest into two subdomains, separated by an interface. A PDE representation is used in one subdomain, and a compartment-based method in the other. These subdomains are labelled Ω_{P} and Ω_{C} respectively. Within the PDE subdomain, the solution is evolved using the Crank-Nicolson method (a finite-difference approximation to the underlying PDE) with zero flux boundary conditions at both ends. The time-step used for the numerical solution of the PDE is Δt and the spatial step is h_p. The compartment based regime is evolved according to the Gillespie SSA, where the subdomain is split into K separate compartments, each of width h_c, so that $|\Omega_{\mathrm{C}}| = Kh_c$. The authors choose $h_c = n_p h_p$ where $n_p \in \mathbb{N}$ is the factor by which the PDE grid is finer than the compartment size. Again, a zero-flux boundary is used within Ω_{C} at the exterior boundary of the subdomain (i.e. the

propensity for jumping out of the domain at that end is set to zero). The zero-flux boundaries on the PDE side of the interface ensure that no mass can leak from one subdomain to the other. The coupling is completed through the use of a pseudo-compartment, C_{-1}. This is a compartment of width h_c adjacent to the interface within Ω_{P}. A schematic for this method is shown in Figure 1.

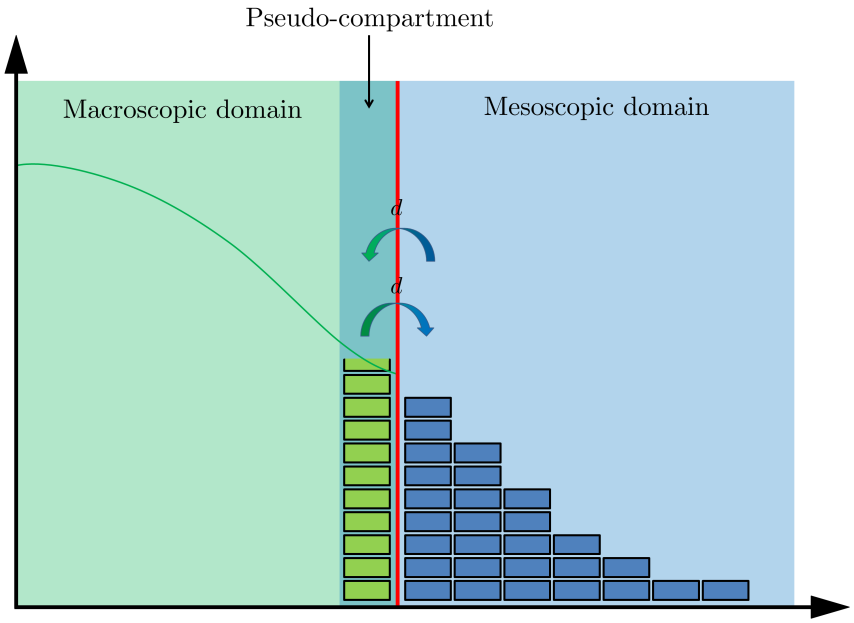


Figure 1. A schematic for the PCM (Yates and Flegg, 2015). The green line represents the PDE solution, while the blue boxes represent particles within each compartment. The red line denotes the interface between the two subdomains. The green boxes represent the number of pseudo-particles within the pseudo-compartment, calculated by direct integration of the solution over that region. The arrows in the centre represent the movement of pseudo-particles over the interface between the pseudo-compartment and the first compartment of the mesoscopic domain.

Pseudo-particle numbers within this pseudo-compartment are calculated through direct integration of the PDE, giving

$$n\left(C_{-1},t\right)=\int_{C_{-1}}c(x,t)\;dx,$$

where $n(A,t)$ is the number of particles residing in the region $A\subseteq\Omega$ at time t. This value is then used to generate a propensity function for particles jumping out of the pseudo-compartment and into the first compartment adjacent to the interface in Ω_{C}. Similarly, in order to correctly model the flux over the interface, particles in the first compartment in Ω_{C} can jump into the pseudo-compartment with the usual diffusive rate.

The algorithm proceeds by firstly generating a time until the next event (a diffusive jump between (pseudo-)compartments or one of the M reactions within the true compartments) according to the Gillespie algorithm (Gillespie, 1977). This can be found by transforming a uniform random variable $u_1\sim\mathrm{Unif}(0,1)$ into an exponential random variable with rate equal to the sum of all propensity func-

tions, given by

$$\tau = \frac{1}{\alpha_0} \ln\left(\frac{1}{u_1}\right), \tag{3}$$

where α_0 is the sum of all propensity functions (including the extra ones for jumps out of and into the pseudo-compartment). The algorithm then checks to see whether the time has been incremented past the next PDE update time. If not, a compartment-based event occurs first, and an event is selected with probability proportional to its propensity function. Otherwise, the numerical solution of the PDE is incremented by a single time-step. When a particle jumps from the pseudo-compartment to the first compartment of Ω_{C}, we remove a particle's worth of mass uniformly from the PDE solution at the points within the pseudo-compartment, and increment the count of particles in the first compartment. A movement in the opposite direction is completed in a similar manner, by adding a particle's worth of mass to the PDE solution uniformly across the pseudo-compartment, and removing a particle from the first compartment. Pseudocode for this method is given in Algorithm 1.

Algorithm 1: Pseudo-compartment method (PCM)

(1a) Initialise the time, $t = t_0$ and set the final time, T. Specify the PDE-update time-step Δt and initialise the next PDE time-step to be $t_\Delta = t + \Delta t$.

(1b) Initialise the number of particles in each compartment in Ω_{C}, $n(C_i, t)$ for $i = 1, \ldots, K$ (where C_i is the region of the domain covered by compartment i), and the distribution of density in Ω_{P}, $c(x, t)$, for $x \in \Omega_{\mathrm{P}}$.

(1c) Calculate the propensity functions for diffusion between the compartments as $\alpha_{i,j} = n(C_i, t)D/h_c^2$ for $i = 1 \ldots K$ and $j = M + 1, M + 2$ (corresponding to left and right movements) and for reactions as $\alpha_{i,j}$ for $i = 1 \ldots K$ and $j = 1, \ldots, M$ using the usual mass action kinetics.

(1d) Calculate the propensity function for diffusion from the pseudo-compartment, C_{-1}, in Ω_{P}, into the adjacent compartment, C_1, in Ω_{C}: $\alpha^* = D \int_{C_{-1}} c(x, t)\, \mathrm{d}x/h_c^2$.

(1e) Calculate the sum of the propensity functions, $\alpha_0 = \sum_{i=1}^{K} \sum_{j=1}^{M+2} \alpha_{i,j} + \alpha^*$.

(1f) Determine the time for the next 'compartment-based' event, $t_c = t + \tau$, where τ is given by equation (3).

(1g) If $t_c < t_\Delta$ then the next compartment-based event occurs:

(a) Determine which event occurs according to the method described in the text (see Gillespie (1977)).

(b) If the event corresponds to $\alpha_{i,j}$ for $i = 1 \ldots K$ and $j = M + 1, M + 2$ then move a particle from interval i in the direction specified by j. If the particle crosses the interface into pseudo-compartment, C_{-1}, then add a particle's worth of mass uniformly to the region C_{-1} i.e. $c(x, t + \tau) = c(x, t) + \mathbb{1}_{[x \in C_{-1}]}/h_c$. Here, $\mathbb{1}_{x \in A}$ is an indicator function which takes the value 1 when $x \in A$ and 0 otherwise.

(c) If the event corresponds to propensity function α^* and $c(x, t) > 1/h_c$ for all $x \in C_{-1}$ then place a particle in C_1. Remove a particle's worth of mass from the PDE solution in the region C_{-1} i.e. $c(x, t + \tau) = c(x, t) - \mathbb{1}_{[x \in C_{-1}]}/h_c$.

(d) Update the current time, $t = t_c$.

(1h) If $t_\Delta < t_c$ the the PDE regime is updated:

(a) Update the PDE solution according to the numerical method.

(b) Update the current time, $t = t_\Delta$ and set the time for the next PDE update step to be $t_\Delta = t_\Delta + \Delta t$.

(1i) If $t \leq T$, return to step **(1c)**.

Else end.

In Figure 2 we have reproduced an example simulation from (Yates and Flegg, 2015) using the pseudo-compartment method. We initialise $N = 500$ particles uniformly throughout the PDE subdomain, where $\Omega_{\text{P}} = (-1, 0)$ and $h_p = 0.01$. The compartment-based subdomain, $\Omega_{\text{C}} = (0, 1)$, is split into $K = 20$ compartments, each of width $h_c = 0.05$. The interface naturally lies at $I = 0$ and the results were averaged over 5000 repeats until a final time of $T = 100$. We set the diffusion coefficient to be $D = 0.0025$ and the PDE time-step to be $\Delta t = 0.01$.

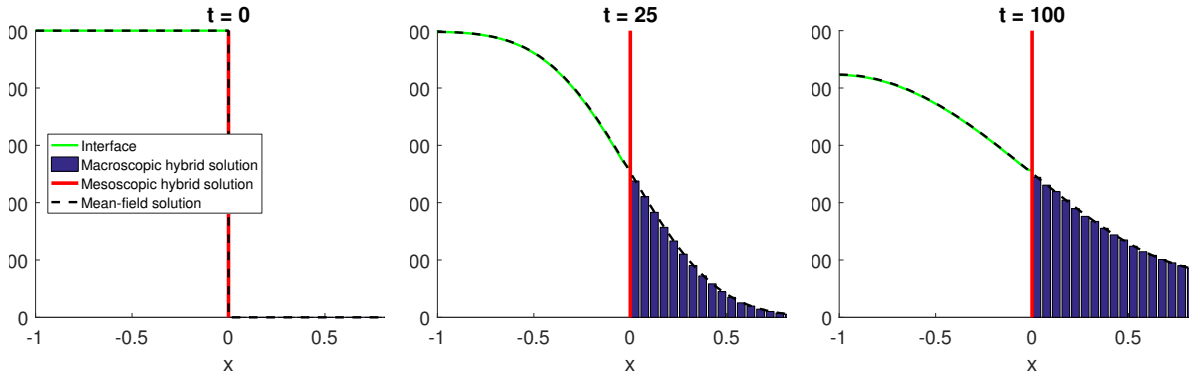


Figure 2. A replication of results from Yates and Flegg (2015) using the PCM. The green line corresponds to the PDE part of the hybrid solution, the red line is the interface, the blue bars are the compartment-based part of the hybrid solution. The dashed black line is the analytical solution of the mean-field PDE model (the diffusion equation) across the entire domain. Parameter values are as in the text.

3.2 Other macro-meso hybrid methods

We now turn our attention to other macro-meso hybrid methods, indicating where they share similarities with one another and where they differ. The full list of methods considered in this section is given in Table 2.

Another type of hybrid method incorporates an adaptive interface. The interface between two modelling regions moves adaptively based on a pre-determined criteria, that may involve (local) copy numbers or densities. Moro (2004) present one such hybrid method when investigating pulled fronts in a diffusive reversible dimerisation. In contrast to the PCM above, they use the same discretisation for both

the continuum and the compartment-based simulations. The boundary between the two subdomains is determined using a threshold number of particles. Any voxels with more particles than this threshold will be simulated by numerically solving the macroscopic Fisher-Kolmogorov-Petrovsky-Piscounov (FKPP) equation. Any voxels with fewer than this number of particles are simulated as a mesoscopic compartment-based position-jump Markov chain. If particles in the compartment-based region jump into the macroscopic region, they are immediately removed from their voxel and held until the next PDE update step. When the PDE update occurs, PDE voxels away from the interface are updated according to the usual finite-difference method, but the value of the voxel closest to the interface is updated with a mixed flux condition. Flux from the macroscopic side to the mesoscopic side is specified by the deterministic flux from the PDE region, whereas flux from the mesoscopic side to the macroscopic side is determined by the number of particles that jumped beyond the interface into the macroscopic subdomain from the mesoscopic subdomain during the PDE update time-step. Flux in the opposite direction (from macroscopic to mesoscopic) is implemented by adding a Poisson distributed random number of particles (with mean corresponding to the expected flux of particles over the boundary as determined by the deterministic model) to the first voxel in the mesoscopic region.

Building upon this idea of adaptive interfaces, Spill et al. (2015) include the possibility of having multiple adaptive interfaces (see Figure 3 for a schematic with a single interface). As in Moro (2004), the same grid spacing is used for both modelling paradigms. The authors are able to add multiple interfaces by again introducing a threshold value in order to determine which regions of the domain should be simulated deterministically and which stochastically, allowing the positions of the interfaces between distinct modelling regions to move, appear and disappear. Boxes with particle numbers lower than the threshold are simulated according to the compartment-based dynamics. Boxes with particle numbers greater than the threshold are categorised as deterministic and evolve according to a set of coupled ODEs which describe the mean field number of particles in each compartment. The single threshold value potentially gives rise to multiple distinct regions of stochastic and deterministic modelling for species whose values fluctuate around the threshold value. In order to ensure there are not too many distinct regions a minimum subdomain size condition is implemented which prevents the occurrence of small, disconnected regions of a particular method.

To implement the coupling between the macroscale and mesoscale models, flux from the deterministic side is governed by the mean-field ODEs, while particles can jump into and out of the interface compartment from the mesoscopic side with rates determined by the SSA (Gillespie, 1977) (in a method similar to that of the PCM (Yates and Flegg, 2015)). All reactions within the interface compartment are completed using the SSA, whereas reactions in other parts of the domain are implemented according to their respective modelling paradigm.

Although many hybrid methods are designed for simulating reaction-diffusion systems, others have been designed to represent different physical phenomena. Schulze et al. (2003) present a hybrid method for modelling epitaxial growth. The method couples a discretised version of the macroscopic Burton-Cabrera-Frank (BCF) continuum model for the growth of a crystalline structure to its corresponding, on-lattice, mesoscopic kinetic Monte-Carlo (KMC) representation. In this mesoscopic model, crystals grow layer upon layer. Layers are first nucleated and then expand by the addition, surface diffusion, and deposition of adatoms (crystalline particles) from solution. The front of a growing layer is referred to as a "step". The method for simulating the KMC model is taken from (Bortz et al., 1975), however, it proceeds in the same way as the Gillespie SSA (Gillespie, 1977). The BCF model, as implemented in this paper, is effectively a finite-difference discretisation of the diffusion equation. This continuum representation is employed in cells which comprise multiple sites of the individual-based model. Steps are simulated using the fine-grained KMC algorithm, and regions away from steps are simulated using the coarse diffusion approximation for the movement of adatoms on the surface. Separating the subdomains are interfaces, which adaptively move with the locations of the steps. The authors consider both two- and three-dimensional simulation regions, referred to as the 1+1- and 2+1-dimensional domains (the "+1"

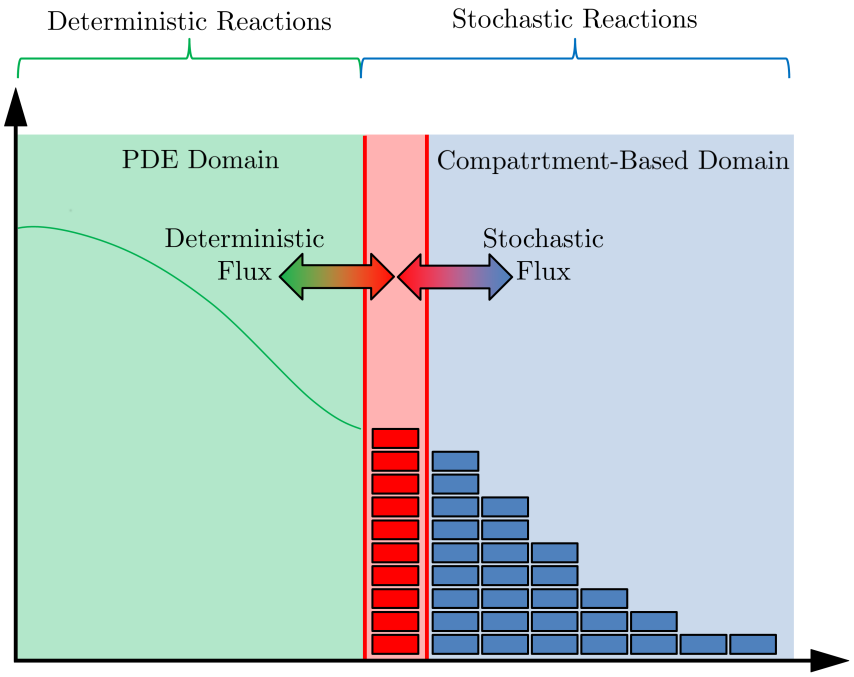


Figure 3. A schematic for the method from Spill et al. (2015). The green line and blue boxes are as in Figure 1, while the red boxes denote an extra compartment between the PDE and compartment subdomains. The coloured double-headed arrows denote how the flux over each of the two red interfaces are calculated.

refers to the crystals growing upwards, meaning that we are effectively simulating a surface process in one- and two-dimensional space).

The algorithm proceeds in a similar way to the PCM (Yates and Flegg, 2015) for reaction-diffusion systems. Close to a step, adatoms are represented using the stochastic KMC algorithm so that their locations can be individually updated, and processes such as absorption, dissociation and nucleation can be accurately modelled. Further away from a step, we neglect these processes and simply consider the particles diffusing along the surface. The time until the next KMC event is calculated using exponentially distributed random variables. If the next KMC event occurs before the next PDE update time, the corresponding event is enacted, otherwise the PDE is evolved forwards in time. Particles jump across the interface, with a rate which depends on the number of particles within the continuum cell adjacent to the interface. These stochastic jump events are simply added to the list of KMC events. If a particle leaves the continuum cell, a new particle is initialised in an adjacent KMC site and the density in the continuum cell is decreased uniformly across its width by a total of one particle. In the opposite direction, the particle is removed from the KMC simulation and a particle's worth of mass is added uniformly across the corresponding continuum cell. As with the PCM, care has to be taken to ensure positive density in the continuum at all times. The interface is also adaptive in that it can evolve as the steps move through space. If a cell needs to change representation from KMC to BCF, we simply count the number of particles in this region and convert it to a particle density uniformly spread across the now-continuum cell. In the opposite direction, the density is converted to the floor of the number of particles (whilst remembering the fractional part in case the cell is again represented by the continuum description later

in the simulation). This number of particles is then initialised randomly throughout the now-discretised cell.

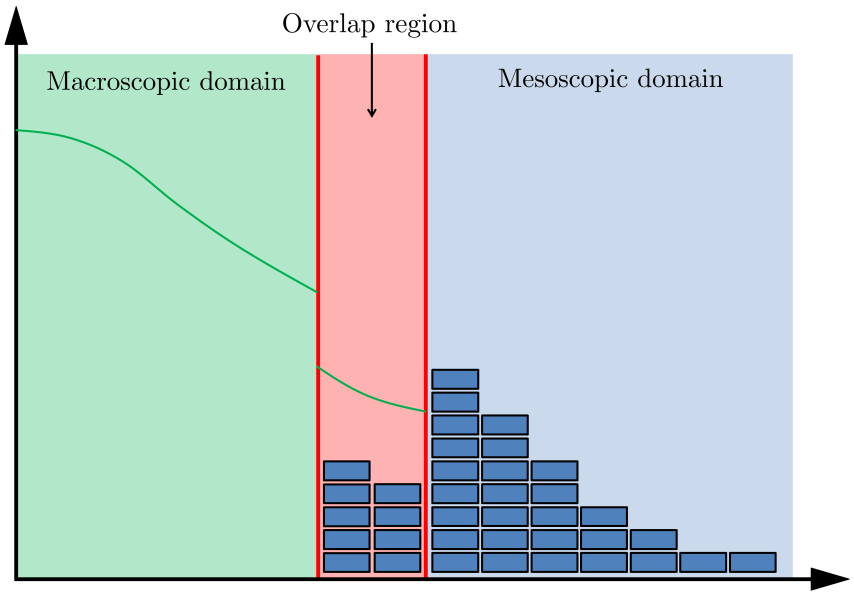


Figure 4. A schematic for the method of Harrison and Yates (2016). The descriptions for the green line and blue bars are the same as in Figure 1. The overlap region is denoted by the red region. The width of the overlap region can be any integer number of compartment widths (here, for simplicity, we have chosen a two compartment-width overlap region). In the overlap region, the sum of the densities of the two methods gives the overall solution.

Point interfaces are not the only way to divide the domain between modelling paradigms – overlap regions may also be employed. Typically these regions inherit properties from both of the models that are being coupled. Harrison and Yates (2016) utilise such a region to couple their mesoscopic and macroscopic models of reaction diffusion. The authors suggest a fixed-time-step, finite-difference scheme for the numerical solution of the macroscopic PDE and use a time-driven algorithm for simulating the stochastic regime (with the same fixed time-step as the PDE). This is in contrast to many of the other hybrid algorithms within this review, in which the Gillespie SSA (Gillespie, 1977) is employed for the mesoscopic regime. It is noted, however, that event-driven alternatives can be applied with minor alterations.

The authors focus on reaction-diffusion systems in one dimension with the compartment-based subdomain on the right and the PDE subdomain on the left (see Figure 4) (although the algorithm would work equally well in higher dimensions and with the orientation of the regions reversed). The overlap region has two interfaces, one at either end. At the right-hand interface where the PDE begins (part-way into the compartment subdomain), a Dirichlet matching boundary condition is implemented on the PDE. This is achieved by calculating the average concentration in the two compartments either side of the interface, and ensuring that the PDE solution at the interface is set to that value. At the left-hand interface, where the compartment-based subdomain ends (part-way into the PDE subdomain), a flux-matching boundary condition is applied to the compartment immediately to the right of the interface. The diffusive flux

across the interface is calculated using the value of the PDE lattice sites corresponding to the centres of compartments either side of the interface. This flux is then imposed on the compartment-based regime by adding or removing particles from the left-most compartment with probability proportional to the magnitude of the flux (with time-step chosen to ensure this magnitude is less than one). An adaptive interface condition similar to that implemented in the adaptive two-regime method (Robinson et al., 2014) (see Section 4.2) is also presented. Repositioning criteria based on density are checked at pre-defined time-steps, and the overlap region is moved accordingly.

Similarly to Harrison and Yates (2016), Flekkøy et al. (2001) utilise an overlap region as part of a non-adaptive algorithm. They introduce a method for coupling a discretised version of the diffusion equation with a discrete-time and -space mesoscopic Markov chain representation of diffusion in which particles can jump to neighbouring voxels in each fixed time-step. The PDE time-step is chosen to be coarser than its stochastic counterpart, meaning that there can be multiple stochastic jumps for every PDE update step. The spatial-mesh for the mesoscopic, stochastic representation is also finer than that of the corresponding discretisation of the diffusion equation; that is to say that there are multiple mesoscopic voxels for every macroscopic voxel. This is in contrast to many of the other macroscopic-to-mesoscopic coupling methods we have outlined in this review, in which the PDE mesh is at least as fine as the compartment size. In these papers, this finer macroscopic resolution was motivated by the idea that the PDE is an exact representation of the scaled probability density of diffusing particles and so warranted an appropriately fine discretisation. Here, Flekkøy et al. (2001) motivate their choice of discretisation (multiple mesoscopic voxels for every macroscopic voxel) by arguing that the PDE-based model is a coarse-grained version of the particle model and hence requires a coarser discretisation in both space and time.

In order to couple the two methods, Flekkøy et al. (2001) allow the two subdomains to overlap across several PDE sites. Within this overlap region, mass is represented as both mesoscopic and macroscopic. The regimes are coupled using a flux-balancing argument which implements the flux of the macroscopic representation on the mesoscopic model at one end of the overlap region and vice versa at the other. The flux term from the PDE description is implemented as a source term which is added to the particle description on the penultimate mesoscopic mesh point. This PDE flux is calculated by using a centred finite-difference approximation across the two PDE sites which span the penultimate mesoscopic mesh point. However, in order to prevent discontinuities in density between the different descriptions, the *PDE* density at one of the two mesh points (used in the finite-difference approximation of the PDE gradient) is substituted for the *particle* density at the same point. At the other end of the overlap region, the averaged particle flux (determined to be the difference between the number of right moving and left moving particles) over a PDE time-step is added to the penultimate site of the PDE mesh.

The previous six methods detailed in the macro-meso section (Flekkøy et al., 2001; Harrison and Yates, 2016; Moro, 2004; Schulze et al., 2003; Spill et al., 2015; Yates and Flegg, 2015) are all spatially-coupled hybrid methods – methods that split the spatial domain into distinct (possibly partially overlapping) regions in which different modelling methods are used. However, other methods exist, which do not specify distinct or even overlapping subdomains for each of the two methods to be coupled. We now focus on two other types of hybrid method. The first employs operator splitting - a process in which the operators which evolve the system are implemented separately (Lo et al., 2016; Rossinelli et al., 2008). The second method employs propensity-based spatial splitting (Chiam et al., 2006), which divides the representation of the dynamics adaptively according to the value of each event's propensity function.

Rossinelli et al. (2008) use τ-leaping (Gillespie, 2001) in order to introduce two new methods for accelerating stochastic reaction-diffusion systems (Cao et al., 2006). The spatial domain is discretised into a regular lattice, with the particles situated at each lattice site subject to the same reactions. Particles can also diffuse to neighbouring lattice sites with appropriately chosen rates.

The first accelerated method presented by Rossinelli et al. (2008) is a purely stochastic algorithm that the authors name the "spatial τ-leap" ($S\tau$-leap) method. This is not a hybrid method, but does allow for faster approximate simulations by employing τ-leaping. This algorithm proceeds by calculating

maximum acceptable leap times for reactions and diffusive events across all voxels. The minimum of these adaptively chosen, acceptable times, τ, is then selected as the next time-step for the algorithm. The entire system is updated by drawing Poisson random variables to simulate the number of events of each type that occur during the next τ time units.

The second method Rossinelli et al. (2008) introduce is the "hybrid τ-leap" ($H\tau$-leap) method. This method exploits the premise that diffusion processes are typically up to two orders of magnitude faster than corresponding reaction processes (Bernstein, 2005). For this method, the authors split the dynamics, completing the diffusive jumps deterministically and the reactions using the τ-leaping method. The time-step for the reactions is calculated adaptively, as before, but only the reactions are updated in this step. Following this, a centred finite-difference approximation combined with forward Euler time-integration is used to deterministically advance the diffusion of particles according to the macroscopic diffusion operator.

A similar operator splitting method is presented by Lo et al. (2016). Their method simulates all reactions using a compartment-based mesoscopic representation, implemented using the Gillespie SSA (Gillespie, 1977). Where molecule numbers are sufficiently large, the number of diffusive jumps between compartments are approximated using continuous Gaussian random variables, with time-dependent means and variances. Where particle numbers are low, diffusive jumps are implemented as events within the SSA. This coupling allows for large time-steps to be taken, even in the presence of rapid diffusion. The numbers of diffusive jumps between compartments are approximated as the sum of the "deterministic" number of jumps and appropriately scaled zero-mean Gaussian random variables. The system size expansion is applied to the reaction-diffusion master equation (RDME) in order to characterise the covariances of these random variables.

Another type of hybrid method chooses which events of the compartment-based regime are to be simulated using the continuum or mesoscopic solvers by using their propensity functions. Chiam et al. (2006) simulate the mesoscopic dynamics using the Gillespie SSA (Gillespie, 1977) while the PDE is discretised using a second-order finite-difference approximation and evolved using the forward Euler method. Each of these descriptions is simulated on the same discretised mesh. Propensity functions are calculated for all possible events (reactions within and diffusive jumps from each box). A threshold value is then used to decide which events are to be simulated using the SSA and which using the deterministic description. The threshold value corresponds to a given fraction of the maximum propensity function. Any events with a sub-threshold propensity are simulated using the SSA. Those with super-threshold propensities are simulated using the finite-difference discretisation. The authors comment that the value of the threshold needs to be "tuned" depending on the specific problem to obtain the correct balance between efficiency and accuracy.

In this section we have outlined several spatially-extended hybrid methods which can be used to couple macroscopic and mesoscopic methods. We now turn our attention towards mesoscopic-to-microscopic couplings.

4 Mesoscopic-to-microscopic models

In this section we will begin by introducing, in broad terms, models which couple microscopic dynamics to mesoscopic dynamics, which we will refer to as "meso-micro" hybrid methods. After summarising the key properties of the meso-micro hybrid methods covered in this section, in Table 3, we go on to describe them in more detail. We begin by giving a detailed description of an illustrative example of a meso-micro hybrid method, the ghost cell method (Flegg et al., 2015) and present pseudocode for its implementation. We then summarise other existing meso-micro hybrid methods.

For meso-micro hybrid methods, both of the models which comprise the hybrid method incorporate some form of stochastic variation. These types of method will be required whenever fluctuations are deemed important across the entire domain, but where specific particle locations are not required in some

subregions of the domain. As an example, we can consider the modelling of an ion channel (Dobramysl et al., 2015; Flegg et al., 2013). We require detailed knowledge of the molecules in regions of space close to the ion channel's receptors in order to resolve the binding dynamics accurately. However, away from the channels, this detailed representation is not required.

Paper	**Type**	**System modelled**
Flegg et al. (2015)	Spatially-coupled, non-adaptive, non-overlap	Reaction–diffusion
Flegg et al. (2012)	Spatially-coupled, non-adaptive, non-overlap	Reaction–diffusion
Robinson et al. (2014)	Spatially-coupled, adaptive, non-overlap	Reaction–diffusion
Flegg et al. (2014)	Spatially-coupled, non-adaptive, non-overlap	Reaction–diffusion
Dobramysl et al. (2015)	Spatially-coupled, non-adaptive, non-overlap	Reaction–diffusion
Hellander et al. (2012)	Operator splitting	Reaction–diffusion
Klann et al. (2012)	Operator splitting	Reaction–diffusion

Table 3. A summary of the meso-micro hybrid papers that will be covered in this section. The methods in all the meso-micro hybrid papers summarised here are designed for modelling reaction-diffusion systems. Each of these papers are concerned with the development of a novel hybrd method, apart from the paper by Dobramysl et al. (2015), which employs the two-regime method (Flegg et al., 2012) to investigate the formation of calcium puffs. See text for more information. Descriptors are as in Table 2.

4.1 Illustrative example of a meso-micro hybrid – the ghost cell method

As an illustrative example for the mesoscopic-to-microscopic methods, we look at the ghost cell method (GCM), developed by Flegg et al. (2015). The domain is divided into two subdomains, which we refer to as Ω_{C} and Ω_{B}, within which the system is evolved according to a compartment-based method and Brownian dynamics respectively. As in the PCM (see Section 3.1), Ω_{C} is split into K compartments of width h_c, so that $|\Omega_{\mathrm{C}}| = Kh_c$. In the Brownian subdomain, particles move in continuous space and a reflective boundary is enforced at the interface to prevent individual particles from entering the compartment-based region due to Brownian jumps. In order to allow the particles to move between the two subdomains, the authors construct a "ghost cell" in Ω_{B} adjacent to the interface with Ω_{C}, which is the same width, h_c, as the compartments. We present a schematic for this method in Figure 5.

Particles move across the interface in both directions according to compartment-based dynamics, with the ghost cell constituting an extra compartment. In order to calculate the propensity function for particles to jump out of the ghost cell, the number of particles in that region of space is simply counted and multiplied by the compartment-based jump rate, d. The Brownian dynamics are implemented with a time-based algorithm and the compartment-based dynamics with an event-driven algorithm. At any time point, the time until the next compartment-based event (including jumps out of and into the ghost cell) is found according to (3). It is then determined whether this event takes place before the next Brownian update. If a Brownian update comes first, the Brownian dynamics are evolved within Ω_{B} for a small time interval, Δt according to (2). Otherwise, the mesoscopic event corresponding to the waiting time is determined and implemented. If a jump from the last compartment to the ghost-cell is enacted, a single particle is removed from the final compartment and is initialised with position chosen uniformly at random across the ghost cell. For movement across the interface in the opposite direction, one of the

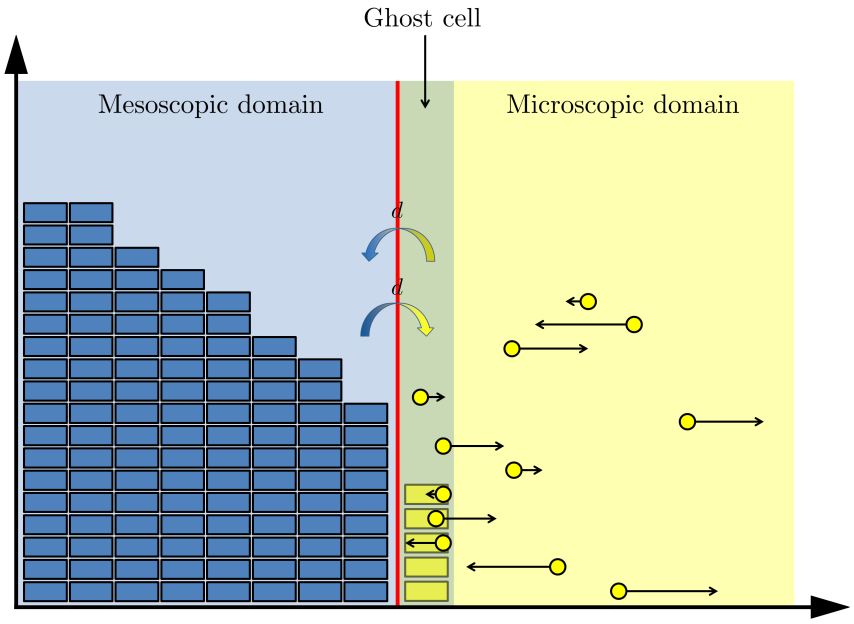


Figure 5. Schematic for the GCM (Flegg et al., 2015). The blue boxes represent particles within each compartment and the yellow dots represent individual particles. These particles are shown with a volume, but in the simulations do not have a mass or volume. The particles reside on the one-dimensional line, but have been illustrated in the plane in order to show the directions and magnitudes of their next movement clearly (black arrows). The yellow boxes within the ghost-cell correspond to the number of Brownian particles which reside within it. The coloured arrows in the centre are similar to those in Figure 1.

Brownian particles in the ghost-cell is chosen uniformly at random and removed from the system. An extra particle is then added to the final compartment of Ω_{C}. Pseudocode for the GCM for diffusion only is provided in Algorithm 2.

Algorithm 2: Ghost cell method (diffusion only)

(2a) Initialise time $t = t_0$, set the final time T. Specify the Brownian update step Δt and set the next Brownian update time to be $t_\Delta = t_0 + \Delta t$.

(2b) Initialise particles in the compartments of Ω_{C} and Brownian particles in Ω_{B}.

(2c) Calculate propensity functions for each compartment given by $\alpha_i(t) = dn_i(t) = Dn_i(t)/h_c^2$ for $i = 1, \ldots, K$, where $n_i(t)$ is the number of particles in compartment i at time t. Calculate the propensity function for diffusion from the ghost cell, $\alpha_{\mathrm{GC}}(t) = n_{\mathrm{GC}}(t)D/h_c^2$, where $n_{\mathrm{GC}}(t)$ is the number of particles in the ghost cell at time t.

(2d) Sum the propensity functions to find $\alpha_0(t)$.

(2e) Determine the time τ until the next compartment-based event according to equation (3). Set $t_c = t + \tau$.

(2f) If $t_c \leq t_\Delta$, then the next compartment-based event occurs:

(a) Choose the event with probability proportional to the associated propensity function.

(b) If the event corresponds to a diffusive jump out of the ghost-cell and into the last compartment, choose one particle in the ghost cell at random to remove and place it in the final compartment of Ω_{C}.

(c) If the event corresponds to a particle jumping from the final compartment of Ω_{C} to the ghost cell, remove a particle from the final compartment and add place it with position chosen uniformly at random across the width of the ghost cell.

(d) If the event corresponds to a purely compartment-based event, implement the jump according to the usual compartment-based dynamics.

(e) Update time $t = t_c$.

(2g) If $t_\Delta < t_c$, we update the Brownian system:

(a) Update the positions of all particles using (3).

(b) Complete reactions using an appropriate method (Andrews and Bray, 2004; Erban and Chapman, 2009; Smoluchowski, 1917).

(c) Update time $t = t_\Delta$. Update $t_\Delta = t + t_\Delta$.

(2h) If $t < T$, return to **(2c)**, otherwise stop.

We have replicated some results from Flegg et al. (2015) using the GCM. These are displayed in Figure 6. As in the PCM, we have placed the interface centrally, $I = 0$, with the mesoscopic subdomain at $\Omega_{\text{C}} = (-1, 0)$ and the microscopic subdomain situated at $\Omega_{\text{B}} = (0, 1)$. We set the Brownian update step to be $\Delta t = 0.01$, and all other parameters are the same as the pseudo-compartment simulation.

4.2 Other meso-micro hybrid methods

We now outline the remaining meso-micro hybrid methods summarised in Table 3. Many of these papers are variations of, or applications of, the same method, namely the two-regime method (Flegg et al., 2012). We start by describing this method, and then follow by describing the adaptations and applications. We then consider two further methods, which fall under the operator splitting category (Hellander et al., 2012; Klann et al., 2012).

Some of the authors of the GCM previously developed the two-regime method (TRM) (Flegg et al., 2012) to couple compartment-based and Brownian-based dynamics. The individual particle paths are evolved according to independent Browninan motions, whilst the compartment regime is updated using the on-lattice, event-based next reaction method (Gibson and Bruck, 2000). Flux over the interface from the compartment-based subdomain to the Brownian-based subdomain is implemented using an altered jump rate to ensure that the flux over the interface is consistent with diffusion. If a particle is selected to jump across the interface from the final compartment to the Brownian-based subdomain, a particle is removed from the relevant compartment and placed at a position selected from a normalised error

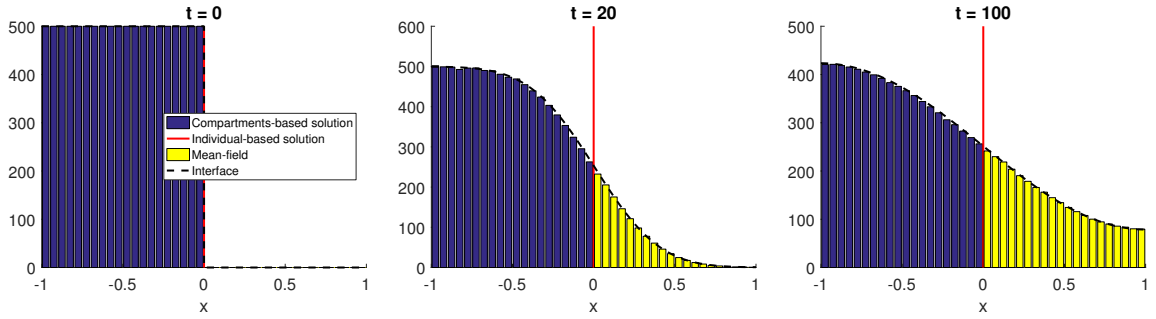


Figure 6. A replication of results from the GMC (Flegg et al., 2015). Descriptions are as in Figure 2, with the addition that yellow bars denote the 'binned' solution of the individual-based simulation in the hybrid method. Parameter values are as in the text.

function probability distribution function. When a particle jumps from the microscopic subdomain to the mesoscopic subdomain, it is simply removed and added to the compartment it has moved in to. The TRM is represented schematically in Figure 7 (a).

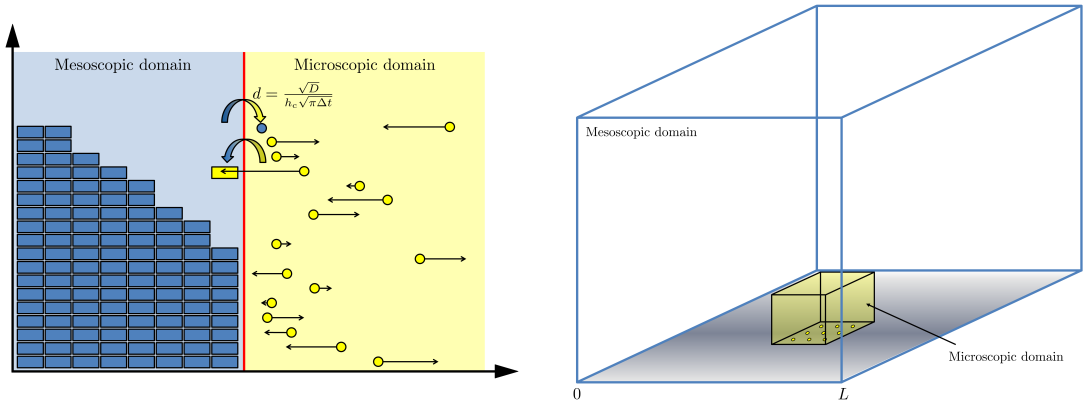


Figure 7. (a) Schematic for the TRM (Flegg et al., 2012). The blue blocks and yellow dots are as described in Figure 5. The arrow from left to right over the interface denotes the jump in this direction, with the specified altered jump rate. In this jump rate, D is the macroscopic diffusion coefficient, h_a is the width of a compartment and Δt is the time-step used to evolve the particles in the Brownian-based subdomain. The other cross interface arrow represents jumps in the other direction. The yellow rectangle and blue particle near the interface represent particles converted from one modelling regime to the other upon crossing the interface in either direction according to the method described. (b) Schematic for the application of the TRM to the problem of calcium-induced calcium release (Dobramysl et al., 2015). The blue outlined box denotes the outer boundaries of the compartment-based subdomain. All boundaries are absorbing, apart from the grey one (bottom), which is reflective. The yellow box in the centre of the lower face is the microscopic subdomain, containing nine ion channels (yellow circles). For simplicity, no particles or compartments are displayed in this schematic.

Robinson et al. (2014) introduce an extension to this method, called the adaptive TRM (ATRM), which adds an adaptive interface to the algorithm. The interface is moved in order to ensure that the subdomain that is to be simulated using the computationally intensive particle-based dynamics is as small as possible. The interface can only move in discrete steps, which are the same size as the width of a compartment in the mesoscopic subdomain. The interface movement condition is, similarly to Moro (2004) (see section 3.2), a local condition. If the number of particles within a compartments width of the interface (and within the microscopic subdomain) is above a pre-specified level, the interface is moved into the microscopic subdomain, extending the mesoscopic subdomain. Conversely, if the number of particles in the compartment adjacent to the interface is below a distinct (lower) threshold, the interface moves towards the mesoscopic subdomain, increasing the size of the microscopic subdomain. The coupling between the compartment-based and Brownian-based methods is implemented exactly as the TRM (Flegg et al., 2012).

The TRM is generalised into two (and higher) dimensions by Flegg et al. (2014). The authors discuss in detail the case of a regular square lattice of points with a planar interface (in which the interface is either purely horizontal or vertical) and cases for which the interface may contain corners. The paper follows a similar method to the TRM paper, in which the authors calculate the factor by which the jump rate over the interface must be scaled by in order for a particle to move from the mesoscopic to microscopic subdomain, together with the rate in the opposite direction.

These methods can be applied to biologically relevant scenarios such as the formation of calcium puffs in a range of eukaryotic cells (Dobramysl et al., 2015; Erban et al., 2014; Flegg et al., 2013). Dobramysl et al. (2015) investigate the formations of such calcium puffs using the TRM. Calcium ions are modelled as diffusive particles, which can bind to activating and inhibiting receptors on the ion channels. Each channel contains four sub-channels, each with one activating and one inhibiting receptor. A sub-channel is activated if the activating receptor has a calcium ion bound to it, and the inhibiting one does not, and a channel is 'open' if at least three of its four sub-channels are activated. When a channel is activated, a constant influx of particles is introduced into the domain. A particle can bind to a receptor with a given probability if it is within a small hemi-sphere of the receptor in question. Particles can also unbind. When particles unbind they are placed a given distance away from the receptor with a second probability. The authors simulate this process in a (three-dimensional) cube representing some part of the cytoplasm of the cell. One face of the cube represents part of the surface of the impermeable endoplasmic reticulum (the cell's major calcium store) upon which a reflecting boundary condition is implemented. In the centre of one of this faces are nine ion channels. On all other faces, an absorbing boundary condition is used. The authors couple the microscopic Brownian dynamics for particle motion in a small cube around the nine ion channels to a mesoscopic compartment-based regime throughout the rest of the domain. The mesoscopic regime is simulated using the next reaction method (Gibson and Bruck, 2000). This hybrid representation is used to investigate calcium puffs which occur when a calcium channel opens and then closes quickly, allowing for a large number of ions to enter the domain over a short time period. This problem is a good example of the need for hybrid methods to couple simulation methods at different scales. If this is simulated using a fully individual-based model, the computational complexity would be too high to simulate accurately within a reasonable time-frame.

Another method which falls into the meso-micro category is presented by Hellander et al. (2012). This is an operator splitting method rather than a spatially-coupled hybrid method. The spatial domain is divided into discrete voxels and the algorithm allows for particular voxels or species to be described as either mesoscopic or microscopic. The algorithm progresses using a splitting scheme. First the microscopic particles are frozen and the mesoscopic particles are progressed using the SSA (Gillespie, 1977). Then, the mesoscopic particles are frozen to allow the microscopic particles to advance according to the Green's function reaction dynamics (van Zon and ten Wolde, 2005). Finally, reactions between mesoscopic and microscopic particles are completed according to the microscopic algorithm, with an adjusted reaction rate to account for the difference in representation.

Operator splitting is also employed by Klann et al. (2012). The spatial domain (assumed three-dimensional) is split into equally sized cubic compartments. Within each of these subvolumes, some species are chosen to be simulated via the compartment-based paradigm using Gillespie's SSA, whilst others are evolved using the Brownian-based approach with a fixed time-step. Thus, different modelling paradigms are used for different species within the same voxel, but also potentially for the same species in different regions of the domain. For each species simulated under the compartment-based paradigm, a minimum time until the next occurrence of *any* type of first-order reaction affecting that species (other than diffusive jumps) is stored. If a particle diffusively jumps out of a compartment (either into a region in which the compartment-based paradigm is being employed for that species or a region in which that species is being modelled as particles) then with probability inversely proportional to the number of particles of its species in the compartment it has just left, the jumping particle takes this minimum first order reaction time with it to the new compartment. The authors use an updated next reaction method (introduced by Anderson (2007)) to implement both reactions and diffusive jumps for particles which are modelled using the compartment-based approach. For particles which are modelled microscopically, diffusion is completed via a discretised SDE which represents Brownian motion, while bimolecular reactions are simulated using the λ-ρ methodology (Erban and Chapman, 2009; Lipková et al., 2011).

If an entire compartment changes description from mesoscopic to microscopic according to the specified criteria, the appropriate number of particles are initialised uniformly throughout the compartment. Of the new individual particles, one inherits the next reaction time for first order reactions from the mesoscopic description, whilst exponentially distributed first reaction times which are later than the inherited time are generated for the others. For a conversion in the opposite direction, the next firing times for diffusive and second- (and higher-, if required) order reactions are calculated according to the standard Gillespie method. For first order reactions, the minimum time (over all the particles of the same species) is used. A similar mechanism is employed if only certain species change their description based on a threshold.

The number of unique methods that we have considered in this category is relatively small. However, the development of the TRM that we have reviewed, serves to demonstrate how a basic method can be altered to incorporate adaptive interfaces and higher dimensions, as well as applied to genuinely multiscale problems. In the following section, we investigate a third category of our spatial coupling involving macroscopic and microscopic models.

5 Macroscopic-to-microscopic methods

In this section, we will introduce and review models which couple macroscopic dynamics to microscopic dynamics, which we will refer to as "macro-micro" hybrid methods. We list and describe the macro-micro hybrid methods covered in this section in Table 4. We begin by summarising an illustrative example of a macro-micro hybrid method, the auxiliary region method (ARM) (Smith and Yates, 2017) and present pseudocode for its implementation. We then summarise other existing macro-micro hybrid methods.

Hybrid methods that couple the macroscopic continuum representations to discrete microscopic dynamics have been relatively poorly studied in comparison to macro-meso and meso-micro hybrid methods. One contributing factor is the fact that such hybrid algorithms bypass the intermediate mesoscale representations of particle dynamics, meaning that the scale separation gap which they must bridge is greater than either of the other two hybrid paradigms. Primarily though, we postulate the relative dearth of macro-micro hybrid methods is due to the inherent difficulty when converting individual Brownian particles into continuum mass (and vice-versa) when coupling individual-based microscopic methods to continuum macroscopic continuum representations.

Although they are less common, macroscopic-to-microscopic methods provide useful insight into a number of biological and physical phenomena, such as the movement of cytochrome c particles in the presence of a charged surface (Gorba et al., 2004).

Paper	Type	System modelled
Smith and Yates (2017)	Spatially-coupled, non-adaptive, no overlap	Reaction-diffusion
Franz et al. (2013a)	Spatially-coupled, non-adaptive, no overlap/overlap	Reaction-diffusion
Geyer et al. (2004)	Spatially-coupled, non-adaptive, no overlap	Reaction-diffusion
Gorba et al. (2004)	Spatially-coupled, non-adaptive, no overlap	Electrostatics
Alexander et al. (2002)	Spatially-coupled, non-adaptive, no overlap	Reaction-diffusion
Alexander et al. (2005)	Spatially-coupled, non-adaptive, no overlap	Viscous gas (train model)
Plapp and Karma (2000)	Spatially-coupled, non-adaptive, no overlap	Dendritic growth

Table 4. A summary of the macro-micro hybrid papers that will be covered in this section. The methods in the macro-micro hybrid papers are designed for modelling a diverse array of applications. Each of these papers are concerned with the development of a novel hybrd method, apart from the paper by Gorba et al. (2004), which uses method they previously developed (Geyer et al., 2004) in order to model the movement of cytochrome c molecules in the presence of a charged surface. Descriptors are as in Table 2.

5.1 Illustrative example of a macro-micro hybrid – the auxiliary region method

As an illustrative example of a macroscopic-to-microscopic hybrid method, we consider the auxiliary region method (ARM) (Smith and Yates, 2017). The ARM couples a PDE for reaction-diffusion systems in a subdomain Ω_{P} to individual-based Brownian dynamics in a subdomain Ω_{B}. Both of the subdomains have zero flux boundaries at the interface so that no PDE mass "leaks" into the individual-based subdomain, and vice versa. Flux over the interface is governed strictly by compartment-based dynamics between the two auxiliary regions, Ω_{PA} and Ω_{BA}, adjacent to the interface within the PDE and Brownian subdomains respectively. The one-dimensional schematic for the ARM is displayed in Figure 8.

In order to implement compartment-based jumps over the interface, particle numbers within each of the auxiliary regions are calculated. For the PDE auxiliary region, the number of auxiliary particles can be calculated as:

$$n_{\text{PA}}(t) = \int_{\Omega_{\text{PA}}} c(x,t)\, dx, \tag{4}$$

where $c(x,t)$ is the solution to the hybrid PDE in Ω_{P}. Similarly, the number of particles within the Brownian auxiliary region is

$$n_{\text{BA}}(t) = |\{j : y_j(t) \in \Omega_{\text{BA}}\}|, \tag{5}$$

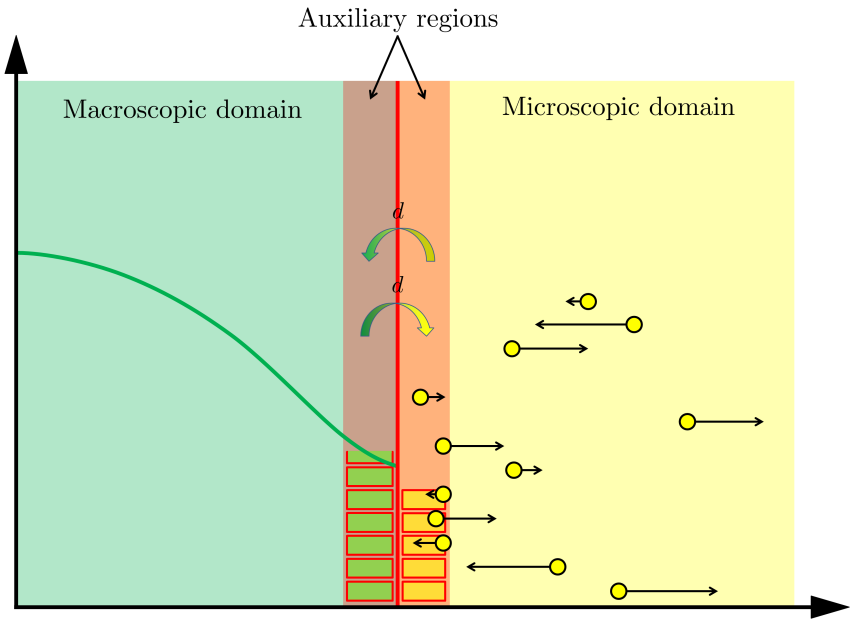


Figure 8. Schematic for the ARM (Smith and Yates, 2017). The green line and yellow dots represent the same phenomena as in Figures 1 and 5 respectively. The auxiliary regions on either side of the interface are highlighted in red. The green and yellow boxes within auxiliary regions represent compartment-based particle numbers in the PDE and Brownian auxiliary regions respectively. The coloured arrows in the centre represent the conversion of particles between the mesoscopic and microscopic auxiliary regions, similar to those in Figure 1.

with $y_j(t)$ the position of particle j at time t. These auxiliary particle numbers are used to calculate propensity functions, which are then employed in an event-driven SSA which determines the time of the next jump across the interface. These auxiliary regions, the dynamics of which are simulated using the compartment-based method, are designed to bridge the gap between the finest and coarsest representations. Particles which jump from the macroscopic subdomain to the microscopic subdomain are removed from the PDE auxiliary region Ω_{PA} by removing one particle's worth of mass uniformly over its width, and are then initialised with position chosen uniformly at random within Ω_{BA}, the Brownian auxiliary region. A movement in the opposite direction is completed by first choosing a particle in Ω_{BA} uniformly at random, removing it, and then adding a particle's worth of mass to the PDE solution uniformly over the region Ω_{PA}.

Reactions are completed using the appropriate methodology for the subdomain in which they reside, with the exception that for reactions with at least one set of participating particles lying within the Brownian auxiliary region, Ω_{BA}. Firings of the reactions involving these subsets of particles are implemented according to the SSA in order to prevent the potential creation of individual-based particles within the PDE subdomain. Pseudocode for the implementation of the ARM is given in Algorithm 3. For simplicity, we present the algorithm for a single species in one dimension.

Algorithm 3: Auxiliary region method (ARM)

(3a) Initialise time $t = t_0$, set final time T, PDE/Brownian update time-step, Δt, the PDE discretisation grid size, h_p, and the auxiliary region width, h_a. Initialise particles in the PDE subdomain, Ω_{P}, and the Brownian subdomain, Ω_{B}, as required. Calculate the time until the next PDE and Brownian update step $t_\Delta = t + \Delta t$.

(3b) Calculate the number of particles n_{PA} and n_{BA} in the auxiliary regions, using formulae (4) and (5) respectively. Consequently, calculate the corresponding propensity functions, $\alpha_{\mathrm{P}}(t) = dn_{\mathrm{PA}}(t)$ and $\alpha_{\mathrm{B}}(t) = dn_{\mathrm{BA}}(t)$. Calculate propensity functions for any relevant reactions within Ω_{BA}, and finally the sum of all the propensity functions to give α_0.

(3c) Calculate the time, τ, until the next auxiliary region event according to equation (3). Update the auxiliary region time $t_c = t + \tau$.

(3d) If $t_c < t_\Delta$

(i) Draw three random numbers $u_1, u_2, u_3 \sim \mathrm{Unif}(0, 1)$.

(ii) If $u_1\alpha_0(t) < \alpha_{\mathrm{PA}}(t)$ (corresponding to a jump from Ω_{PA} to Ω_{BA}):

- Remove a particle from the PDE auxiliary region according to
$$c(x,t) = c(x,t) - \frac{1}{h_a}\mathbb{1}_{[x\in\Omega_{\mathrm{PA}}]}.$$
- Initialise a new particle of uniformly within Ω_{BA} with position $y^* = u_3 h_a + I$.

Else if $u_1\alpha_0(t) < \alpha_{\mathrm{P}}(t) + \alpha_{\mathrm{B}}(t)$ (corresponding to a jump from Ω_{BA} to Ω_{PA}):

- Choose a particle at random from within the Brownian auxiliary region and remove it from the system by selecting an index q according to $q = \lceil u_3 n_{\mathrm{BA}} \rceil$ (where $\lceil x \rceil$ represents the smallest integer greater than x).
- Add a new particle into the PDE auxiliary region according to
$$c(x,t) = c(x,t) + \frac{1}{h_a}\mathbb{1}_{[x\in\Omega_{\mathrm{PA}}]}.$$

Else (corresponding to a reaction in Ω_{BA})

- Use u_2 to choose a reaction to be implemented from the list of possible reactions with probability proportional to its propensity function.
- Enact the reaction chosen in the previous step according to the usual kinetics of the reaction pathway

(iii) Set $t = t_c$

Else

(i) Update the PDE system using an appropriate numerical method.

(ii) Implement any reactions in Ω_{B} using any appropriate method. Note that production reactions should be implemented after any degradation reactions in order to prevent particles being created and destroyed in the same time-step.

(iii) Update the positions of the Brownian particles according to equation (2), including any boundary conditions

(iv) Set $t = t_\Delta$, update $t_\Delta = t + \Delta t$.

(3e) If $t < T$, return to **(3b)**, otherwise stop.

As with the PCM and GCM, we have replicated some of the results from (Smith and Yates, 2017) using the ARM. For these examples, the macroscopic subdomain is $\Omega_{\text{P}} = (-1, 0)$ and the microscopic, Brownian subdomain is $\Omega_{\text{B}} = (0, 1)$. Both auxiliary regions are set to be size $h_a = 0.05$, and the time-step for both the Brownian and PDE updates are set to $\Delta t = 0.01$. All other parameter values are as in the previous simulations. The results are shown for the same initial condition as in Figure 6.

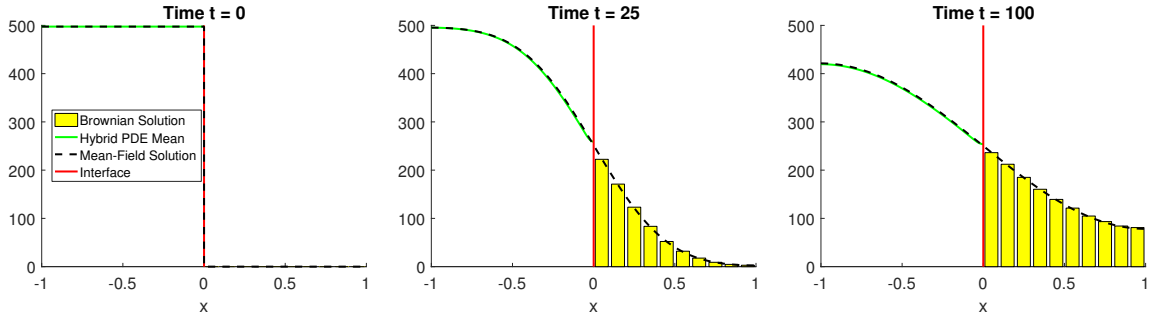


Figure 9. Replication of results from the ARM (Smith and Yates, 2017). Descriptions for the PDE and Brownian domains are as in Figures 2 and 6, respectively, with parameter values given in the text.

5.2 Other macro-micro hybrid methods

Franz et al. (2013a) present a macro-micro hybrid method in which the coupling is completed directly, without the use of a compartment-based intermediary regime (Figure 10). In the microscopic subdomain, particles evolve their positions according to Brownian motion. The corresponding Fokker-Planck equation which describes the evolution of the probability density of each particle is the diffusion equation.

The conversion of PDE mass to individual particles is achieved by allowing PDE mass to flow over the interface and probabilistically determining whether sufficient mass has crossed the interface to warrant the instantiation of a new Brownian particle. Conversely, Brownian particles crossing he interface in the opposite direction are realised as delta function contributions to the PDE solution at the position at which they arrive at the end of their jump (10).

Upon finding that their initial coupling algorithm can correctly maintain mean particle concentrations, but incorrectly matches particle variance profiles, Franz et al. (2013a) adapt their algorithm by incorporating an overlap region in which some of the mass is represented as PDE and some as Brownian particles. At the interface at one end of the overlap region, PDE mass is converted in to particles, as before, and at the the other end, particles are incorporated into the PDE by the addition of delta functions as previously. The addition of this overlap region corrects the variance of the particles in the purely Brownian region of the hybrid simulations.

Geyer et al. (2004) also allow mass from the PDE to flow over the interface. They introduce two methods to interface Brownian dynamics simulations for diffusion to a deterministic macroscopic density-based representation. The first method couples individual particles to a constant density reservoir, whereas in the second, the macroscopic subdomain itself evolves according to a discretised version of the diffusion equation. In the first case, the authors ensure the correct movement over the boundary by removing particles when they cross into the reservoir from the Brownian dynamics subdomain, and inserting new particles into the Brownian dynamics subdomain with an appropriate rate and position. The rate and position are determined by using the fundamental solution of the diffusion equation to calculate the probability density function (PDF) and magnitude of mass which has flowed over the interface in the

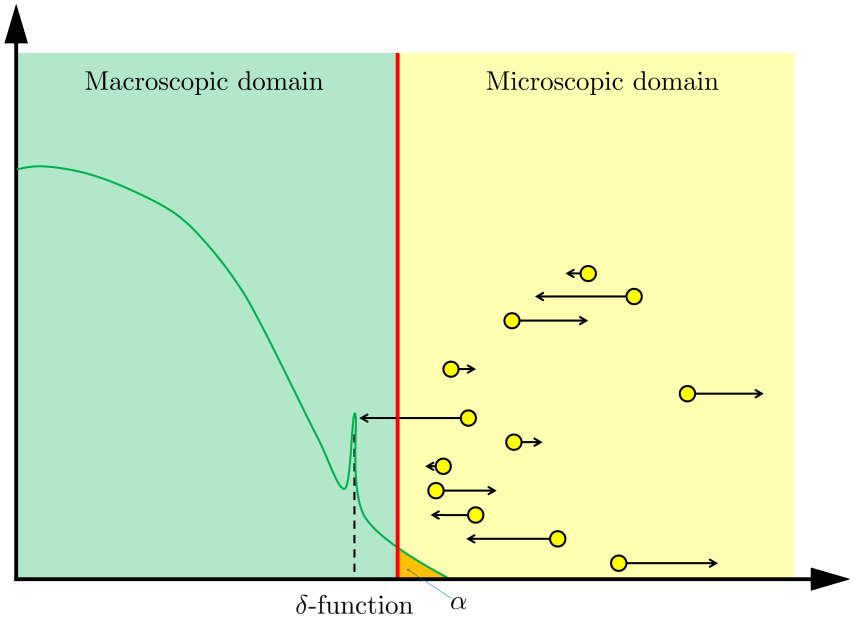


Figure 10. Schematic for the method by Franz et al. (2013a) (without overlap region). The green line and yellow dots represent the same quantities as in Figure 8. The orange mass labelled α is the amount of mass that flows over the interface in a small time-interval (comprising several PDE updates). Its total mass is used to find the probability of a particle being initialised in the microscopic subdomain, and its profile acts as a scaled probability density function for the position of the new molecule. The spike in the PDE solution is representative of a Dirac delta-function which is added to the PDE at the location that a Brownian particle has jumped to from the Brownian subdomain.

intervening time period. This can then be used to determine if, and where, a particle should be placed in the microscopic Brownian dynamics subdomain.

For their second hybrid method (see Figure 11 (a)), which couples Brownian particles to a dynamic PDE, the PDE mesh-point located closest to the interface is used to determine the probability density function of particles flowing into the Brownian subdomain (i.e. it is treated as a constant density reservoir as in the fixed density case). This relies on choosing the PDE mesh width sufficiently large (and thus sacrificing accuracy for the PDE solution) or the time step to be sufficiently small so that the majority of the mass that flows in to the Brownian subdomain originates in this region. However, the value of the PDE solution at this mesh-point is allowed to evolve dynamically according to diffusive fluxes. The flux into this PDE mesh-point from the Brownian dynamics side is proportional to the net number of particles which have flowed between the regions in the preceding time-step. The flux from the remainder of the PDE subdomain is calculated according to the usual centred finite-difference approximation of the diffusion equation.

The first method is then used by Gorba et al. (2004) to investigate the behaviour of cytochrome c molecules which move in the presence of a charged membrane. Two kinds of external force are considered (electrostatic interaction and van der Waals forces) between pairs of cytochrome c molecules and between cytochrome c molecules and the charged membrane. The system is modelled as follows. The region

of interest (see Figure 11 (b)) is a cuboid-shape box, with equal width and length. On each side of of the box, reflective boundary conditions are implemented, whilst the base of the box has a repelling boundary condition due to the repulsion caused by van der Waals forces between the membrane and the molecules. At a prescribed height there is an interface, below which particles evolve according to a Langevin equation, and above which is a fixed-density reservoir of particles. All simulations using this method are initialised with no particles in the Brownian subdomain, with particles entering solely via the reservoir.

The authors compare the results using their hybrid coupling algorithm with previous simulation results, which assume a fixed number of particles with a zero-flux boundary condition replacing the reservoir at the top of the box. They show that the shape of concentration profiles as a function of distance from the membrane generated by the two methods agree.

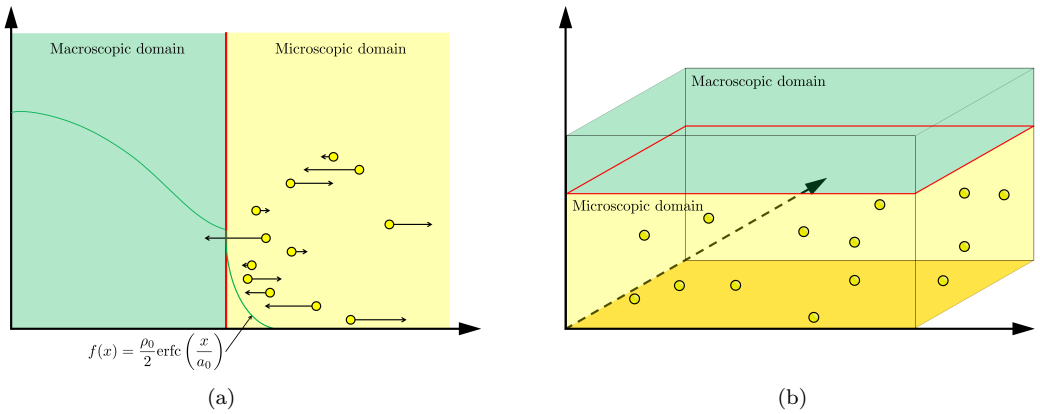


Figure 11. (a) Schematic for the method presented by Geyer et al. (2004). The green lines and yellow dots represent the same phenomena as in Figure 8. The additional green line which resides in the microscopic subdomain is the mass which flows over the interface after a given time, where ρ_0 is the density at the PDE meshpoint adjacent to the interface and $\sigma = 2\sqrt{D\Delta t}$ is the average Brownian step size during a time interval of length Δt. (b) Schematic for the application presented by Gorba et al. (2004). The yellow dots are the same as in Figure 8, while the blue region is a constant density heat-bath. There are reflective boundary conditions on all sides of the computational domain, with the exception of the lower boundary, denoted in orange. This is a repulsive boundary caused by the van der Waals forces, representing the charged boundary.

In contrast to the previous works presented here, Alexander et al. (2002) introduce a hybrid method to couple an *SPDE* (as well as a similar algorithm for a PDE) to Brownian dynamics (see Figure 12). Separating the continuum and individual-based subdomains is an interface, over which particle fluxes are matched to ensure that particle movement is correctly calculated between the two descriptions. The continuum subdomain is divided into a mesh, upon which the solution to the SPDE/PDE is calculated numerically. In the particle-based subdomain, particles move according to the standard off-lattice Brownian motion SDE. The hybrid algorithm progresses in discrete time with both subdomains using the same time-step.

In order to hybridise the two methods, at the beginning of each time-step, an integer number of particles are uniformly initialised within the SPDE/PDE voxel closest to the interface, referred to as the "handshaking" region. The number of particles initialised is the closest integer to the value of the SPDE/PDE solution at the handshaking mesh point at the beginning of the time-step. All particles (both in the handshaking region and elsewhere) are then evolved according to the standard Brownian

motion equation. The number of particles crossing the interface gives the flux into the handshaking mesh point which is stored and later implemented when the PDE/SPDE values are updated. Any particles which do not reside in the Brownian subdomain following the position update step are removed from the simulation. All other SPDE/PDE fluxes are calculated using the discretised version of the SPDE/PDE equation and the values of the mesh points are consequently updated.

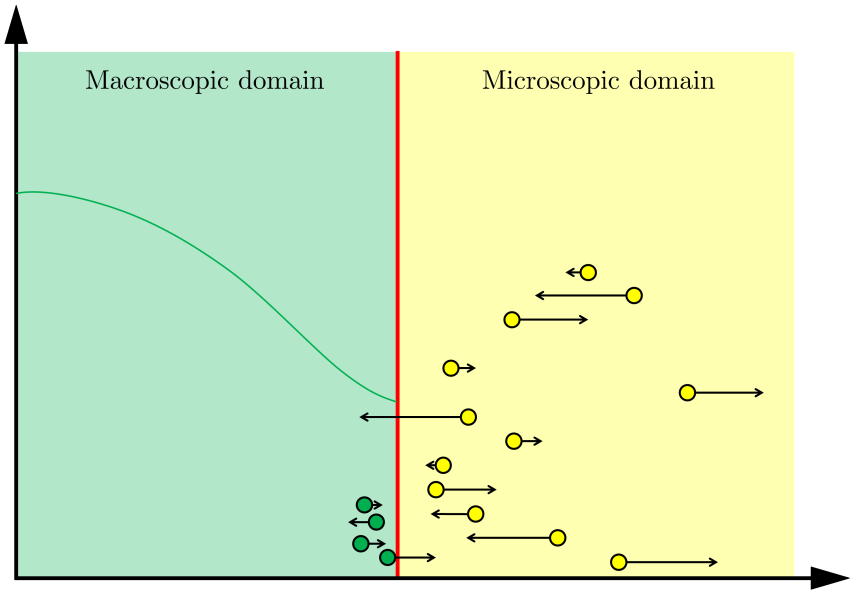


Figure 12. Schematic for the method by Alexander et al. (2002). The green line and yellow dots represent the same phenomena as in Figure 8. The green dots residing within the PDE subdomain are particles initialised at the beginning of a time-step (corresponding to the numbers of particles within the corresponding region obtained by direct integration of the PDE solution). Black arrows show the directions and magnitudes of next movement of all particles. The discretisation on the lower axis is the PDE mesh over the entire domain.

In a later paper, the same authors also consider correlated systems (Alexander et al., 2005). They develop a hybrid algorithm for the train model which describes the transport of material in a viscous gas. This model is chosen due to its relative simplicity and the readily derived continuum (SPDE/PDE) counterparts which are straightforward to solve numerically. The train model can be summarised as follows: several trains run parallel to one another at different speeds with varying numbers of passengers. Passengers jump, with exponentially distributed waiting times, between neighbouring trains, changing the momentum of the participating trains. At each end of the array of trains are "platforms" which move at a fixed velocity and contain a reservoir of passengers.

The authors couple a discretised version of the SPDE/PDE representation of the train model to the discrete individual-based description. Both the discretised SPDE/PDE and the train model are simulated with the same grid spacing. Separating the two subdomains is an interface. The hybrid algorithm uses flux-matching for both the velocity and the momentum over the interface, whilst also maintaining the long-range spatial correlations in the velocity caused by stochastic fluctuations. The algorithm employed

is analogous to the one that is presented in Alexander et al. (2002). At the beginning of a continuum time-step the first voxel in the continuum part of the domain (called the "handshaking" region) is filled with particles. The number of particles initialised is the nearest integer value to the SPDE/PDE solution in this voxel. Each of these particles is also assigned a velocity which corresponds to the velocity of the continuum model at that point. The individual-based particles are then evolved and the fluxes of velocity and momentum over the interface are calculated. These values are then utilised within the continuum solver in place of the the fluxes over the interface.

Finally, Plapp and Karma (2000) introduce a hybrid method for simulating interfacial patterns, with specific application to dendritic crystal growth. In the inner-region, which includes the area in which the crystal is growing and a buffer layer of liquid adjacent the interface, a discretised version of the diffusion equation is solved and the position of the crystal interface is updated using a deterministic phase-field approach. This update method is coupled to particles evolving according to off-lattice Brownian motion. The time-step at which the positions of particles are updated increases the further away the particles are from the interface. At the edge of the inner-region between the crystal surface and the outer region is a "buffer-region" of undercooled liquid. This buffer-region acts to damp the stochastic variation of the outer-region to negligible levels at the crystal surface. Adjacent to the interface between in the inner and outer regions are "conversion cells" which facilitate the conversion of Brownian walkers into PDE density and vice versa, via the implementation of boundary conditions on each of the models. A Dirichlet boundary condition for the PDE is determined by the number of Brownian particles residing in each of the conversion cells. In the other direction, the heat flux over the boundary is collected in a reservoir. If the value of the reservoir exceeds a threshold, H, a new particle is added to the cell. If it drops below $-H$, then a particle is absorbed and consequently removed from the corresponding conversion cell.

6 Other hybrid methods

Within this section, we investigate some other hybrid methods that do not fall within any of the above three categories. The section will encompass microscopic-to-molecular dynamics spatially-coupled methods. These hybrid methods are typically designed to represent hydrodynamical systems, adaptive mesh and algorithm refinement and quasicontinuum methods. We will also investigate another class of hybrid methods, which we shall call "species splitting", where different species are simulated using different representations.

6.1 Micro-molecular methods

In this subsection, we present a paper which introduces hybrid methods for coupling a molecular dynamics model to a corresponding Brownian motion model for the movement of a large particle in a surrounding 'molecular' medium.

Erban (2014) introduces one such spatial hybrid method in one and three dimensions. The author motivates the use of such a method by considering a large focal protein molecule which is being moved by interactions with the smaller water molecules that surround it. The protein molecule is modelled as a hard sphere with a larger radius and mass than the water molecules. The motion of the molecules in this molecular dynamics model are fully deterministic once they have been randomly initialised, with changes in velocity caused by momentum exchange. If the protein molecule were to be modelled using Brownian dynamics or the Langevin equation (respectively), the interactions between it and the surrounding water molecules could be encapsulated implicitly through the random changes in position or velocity (respectively) of the protein. Erban (2014) demonstrates the equivalence between the motion of the protein molecule in the molecular dynamics simulation to the motion specified by the corresponding Langevin or Brownian dynamics equations in certain limits. This equivalence engenders the possibility of a hybrid method.

In both the one- and three-dimensional hybrid methods, the domain is split into two subdomains: one in which water molecules are explicitly simulated and the other in which the water molecules are modelled implicitly and the protein moves according to the appropriate Langevin equation. The first coupling algorithm introduced is for a one-dimensional domain, in which water molecules are initialised across a subset of the real line according to a spatial Poisson point process with a specific density, while velocities are normally distributed with zero mean and variance which incorporates the diffusion coefficient, the ratio between the large and small particles' masses and a friction coefficient. Collisions between water molecules and proteins are elastic and subject to conservation of momentum. Any water molecules which leave the molecular dynamics subdomain are removed from the system. Molecular dynamics particles can also be created towards the edges of the subdomain, and are initialised using a normalised complementary error function. This maintains the density of water molecules in the molecular dynamics heat bath. The three-dimensional algorithm is similar. The algorithms are time-driven, that is the system is evolved by implementing exchange of momentum through collisions, updating positions and the addition and removal of heat bath molecules at each fixed time-step. There is a constraint on the size of the time-step to ensure that at most one macro particle enters the subdomain in each time-step. A similar coupling is presented in Erban (2016).

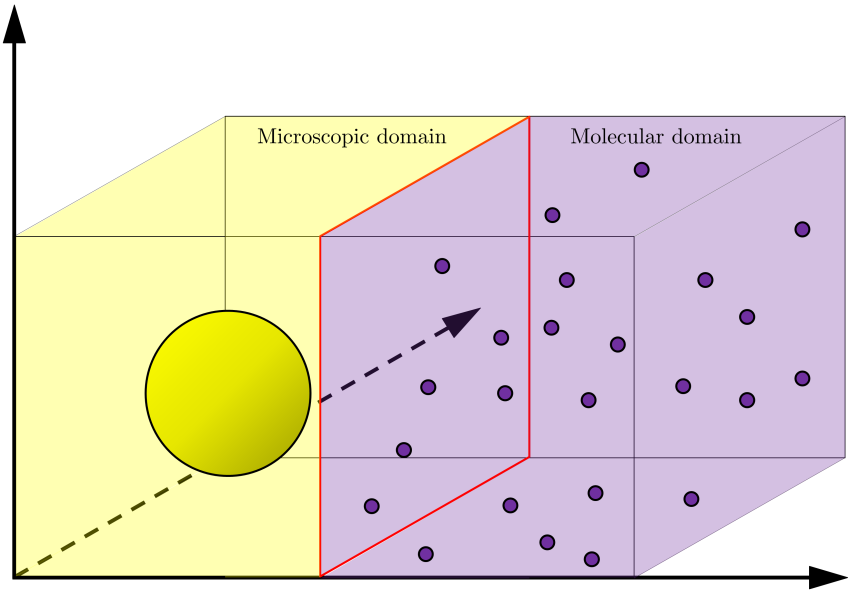


Figure 13. A schematic for the method presented by Erban (2014). The large yellow circle is an individual particle (protein molecule) with mass, volume and velocity. The small purple dots represent the molecular dynamics particles (air/water molecules) and also have a mass, volume and velocity.

6.2 Hydrodynamics

Whilst most of the examples that have been presented in Sections 3-5 are designed to represent reaction-diffusion systems (with noted exceptions), these are not the only systems in which spatial hybrid methods

have been employed. In this subsection we review spatial hybrid methods and their uses in modelling hydrodynamics in an efficient and accurate manner.

The most common type of spatially-coupled hybrid method employed within hydrodynamics is macro-micro couplings. Donev et al. (2010) couple the stochastic hydrodynamics model given by the Landau-Lifshitz Navier-Stokes (LLNS) equations, to a corresponding direct simulation Monte-Carlo representation. The LLNS equations include hydrodynamic fluctuations, and, as such, are SPDEs. They are simulated using a fixed-time, three-stage Runge-Kutta integration scheme (a finite volume method) although the authors note that other other finite-volume explicit schemes can be substituted. Within the particle subdomain, the hydrodynamics are simulated using a fixed-time stochastic momentum exchange method which preserves the essential hydrodynamic properties of molecular dynamics. The time-scale of the micro solver is smaller than that of the macro solver, so that multiple particle updates occur for every continuum update. This is in contrast to PDE-assisted Brownian dynamics (Franz et al., 2013a) for reaction-diffusion systems which does the opposite.

Within the continuum subdomain, the only quantities that need to be considered are the conserved variables of mass, momentum and energy within each continuum cell, as well as the continuum normal flux between any two neighbouring macroscopic cells. Within the particle subdomain, inter-atomic forces are simulated by stochastic collisions, so that any particles within a given distance have a probability of colliding. Separating the two subdomains is an (adaptive) interface. The coupling algorithm ensures that both the fluxes and the states (density, momentum and energy) at the interface are continuous by introducing a state-flux coupling methodology; the macroscopic LLNS equations act as a source of particles into the microscopic subdomain at the interface, and the particles impose a flux boundary condition on the continuum. To impose the state boundary condition from the continuum subdomain onto the particle subdomain, a reservoir of temporary particles (in a small region within macro cells adjacent to the interface) are initialised (every micro time-step) with some velocity and temperature according to a Maxwell-Boltzmann or Chapman-Enskog distribution chosen to match the velocity and temperature of the associated macro cell (reminiscent of the method of Alexander et al. (2002) for modelling diffusion). The number of these particles is chosen to match the continuum density in the associated macro cell. The particle flux over the interface is calculated and stored every micro time-step and imposed on the continuum solver at the end of every macro time-step.

There are other methods which also utilise an interface in order to couple two subdomains. Flekkøy and Coveney (1999) couple the mesoscopic dissipative particle dynamics to the derived Langevin equation in order to simulate the movement of large colloid molecules. O'Connell and Thompson (1995) also utilise an interface in order to create a generic algorithm for simulating a macroscopic and microscopic representation of a fluid system. The authors couple by averaging the velocities of the individual particles close to the interface, providing a boundary condition for the corresponding continuum model.

Overlap regions have also been employed in the hydrodynamics literature. Flekkøy et al. (2000) couple a macroscopic PDE to a microscopic method in which particles interact according to Lennard-Jones potentials (Allen and Tildesley, 2017). Separating the two subdomains is an overlap region in which both the particle and continuum descriptions are valid. The conservation of mass and momentum between the two regions is handled explicitly using flux exchange, which means that the coupling scheme adheres to the relevant conservation laws.

Within the continuum description, the mass and momentum fluxes are represented using finite-differences across each continuum node. These finite-difference approximations are used to advance the continuum equations in time. The boundary conditions derived from the particle region are implemented on the continuum representation by replacing the fluxes at the end of the continuum subdomain with the mean mass and momentum fluxes of particles around the boundary, averaged over a continuum time-step. To implement the fluxes of mass and momentum from the macroscopic to microscopic subdomain, a number of particles per unit time (determined in order to conserve mass flux) are placed into a region close to the boundary of the particle subdomain. Additionally, the velocities of the particles are chosen

to conserve the flux of momentum. The authors note that there is an asymmetry relating to the fluctuations using their method; the continuum subdomain effectively acts to damp fluctuations in the particle subdomain meaning, for example, that fluctuations in particle numbers will be diminished in comparison to predictions from statistical mechanics (reminiscent of the damping of the Brownian dynamics by the PDE observed by Franz et al. (2013a)).

A second coupling, presented by Wagner and Flekkøy (2004) extends previous works (Flekkøy et al., 2000; Wagner et al., 2002), in which fluxes for momentum and mass were preserved between the two subdomains, to the situation in which energy flux is also conserved. The authors also investigate the limitations of this hybrid representation when simulating both homogeneous and gradient flow.

The continuum equations are discretised using a centred finite-difference scheme on a regular mesh. Separating the continuum and particle subdomains is an overlap region which allows for the conservation of flux between the two descriptions. To calculate the continuum flux for the penultimate node within the overlap region (which corresponds to the boundary of the particle subdomain), a similar method to the one employed by Flekkøy et al. (2001) is used. One of the terms in the centred finite-difference approximation is replaced by the corresponding value from the particle subdomain at the particle end of the overlap region. These fluxes (for mass momentum and energy) are then arithmetically averaged with the corresponding mean fluxes of the particles that occupy positions within the final voxel of the overlap region. These mean fluxes are then used to implement Neumann boundary conditions on the final node of the continuum representation. The same averaged fluxes are implemented on the particle subdomain by adding/removing particles to/from the microscopic description in a region corresponding to the penultimate node of the continuum discretisation. To ensure that both momentum and energy are conserved, velocities and accelerations of particles in the overlap region are altered accordingly.

Several other papers have adopted the use of an overlap region. Wagner et al. (2002) use mutual flux exchange in order to couple their finite-difference representation of a PDE for fluid flow to the corresponding microscopic dynamics. The authors measure the fluxes for mass, momentum and energy in order to ensure conservation. Delgado-Buscalioni et al. (2005a) and Delgado-Buscalioni et al. (2005b) present two further papers which couple using flux conservation. These methods use flux exchange from the continuum to particle density in order to modify the microscopic description, while fluxes in the opposite direction supply boundary conditions for the continuum representation.

Delgado-Buscalioni et al. (2009) present a hybrid method with three spatial scales - coupling the macroscopic to the mesoscopic to the microscopic scales, with an application to liquid water. The authors use two different schemes in order to complete the coupling. To couple between the macro and microscales, the HybridMD scheme is used (De Fabritiis et al., 2006) and to couple the microscale to the mesoscale, the adaptive resolution scheme (AdResS) is employed (Praprotnik et al., 2005).

There are many other papers which have addressed hybrid methods for hydrodynamics. We direct the interested reader to the reviews of Koumoutsakos (2005) and Mohamed and Mohamad (2010) and the PhD thesis of Hadjiconstantinou (1999) for further details.

6.3 Adaptive mesh and algorithm refinement

Adaptive mesh refinement (AMR) is a method for evaluating PDE solutions on inhomogeneous domains, in which coarse cells are recursively refined in both time and space in regions of high sensitivity (Berger and Colella, 1989). Adaptive mesh and algorithm refinement (AMAR) extends the idea of AMR. The difference between AMR and AMAR is that when the predefined highest spatial resolution has been reached, AMAR switches to using a discrete method for simulating the underlying phenomena. The coupling between the coarse PDE and the fine discrete method is completed using a buffer region residing within the PDE region close to the interface between the two subdomains. Particles are created within this region at the beginning of the fixed PDE update time-step with the appropriate physical quantities such as mass, momentum and energy, and are then allowed to flow forwards in time. This provides

boundary conditions for the two systems. Garcia et al. (1999) and Williams et al. (2008) use AMAR in order to accurately model hydrodynamic flow.

6.4 Quasicontinuum methods

Quasicontinuum (QC) methods combine continuum and atomistic representations for modelling crystalline structures, and were first introduced by Tadmor et al. (1996). Shenoy et al. (1999) propose a hybrid method for coupling the atomistic-scale dynamics of solid deformation to a corresponding continuum description. The quasicontinuum method exploits the kinematic constraints inherent to the atomistic lattice, reducing the large number of degrees of freedom by employing the finite-element method in order to simplify the minimisation of the potential energy associated with the system under a deformation. The system of interest is typically made up of a huge number of atoms, and consequently has an extremely large number degrees of freedom. It is therefore computationally difficult to calculate any quantity of interest. To reduce the number of degrees of freedom, a subset of the atoms are chosen to be *representative atoms*. Each representative atom is a proxy for a number of neighbouring atoms, reducing the number of degrees of freedom. Close to the deformation, where each atom experiences a different local environment, atoms are represented individually. In these regions, an atomistic, non-linear approach to calculating the energy is required. Further from the deformation, where non-linear effects are negligible and each representative atom is a proxy for some of its neighbours, linear elasticity theory is used. This allows for the faster calculation of the energy landscape in large regions of the spatial domain without the loss of accuracy in the regions in which a more detailed representation is required. The condition which specifies the homogeneity, or otherwise, of a local region is determined by calculating the right stretch tensor of the deformation. If the maximum difference of the eigenvalues over any pair of atoms within a given distance is less than a pre-determined threshold, it is treated as a near-homogeneous environment. This ensures that the algorithm adaptively chooses which regions are to be treated as homogeneous. However, the algorithm does create additional forces, referred to as "ghost forces", due to the hybridisation. These are corrected for by applying correction forces within the energy minimisation calculation.

6.5 Other hybrids

This section contains several hybrid methods that do not fall into the spatially-coupled reaction-diffusion, or hydrodynamics categories. They are designed to model a wealth of different mathematical, biological and physical problems and employ a variety of hybridisation techniques.

Jeschke and Uhrmacher (2008) introduce a hybrid method for the simulation of macromolecular crowding. They combine the mesoscopic next subvolume method (NSM) (Elf and Ehrenberg, 2004) for the efficient simulation of compartment-based reaction-diffusion systems with an off-lattice representation of large crowding particles (crowders). The crowders are spherical and evolve according to an individual-based method which assumes random movements of particles over fixed time-intervals. All other particles are updated using the NSM on a square lattice.

Crowders occupy a certain volume. As they move, the volume that is available for the compartment-based particles and their interactions changes. Any compartments which intersect a crowder are sub-divided, using an octree refinement algorithm, until a pre-defined number of sub-divisions have been completed. The volume of the compartment that is occupied by the crowder is then approximated as the number of sub-octants that intersect the it. The crowders and compartment-based particles can interact with one another. For example, the location of overlapping crowders will influence the neighbouring compartments into which compartment-based particles are able to diffuse. Diffusion occurs at the usual diffusive rate, but scaled down by the proportion of the boundary between the current compartment and the neighbouring compartments that is occupied by crowders. Particles can also bind to the crowders, meaning that they are removed from the NSM reactions list and move about with the crowder. When the crowders move, they "push" the compartment-based particles into the

unoccupied region of their current compartment, or into neighbouring compartments if the crowder completely fills their current compartment. All movements, reactions and steric interactions are controlled by the "coordinator component" which keeps track of all putative next event times, schedules the next reaction and updates the two systems.

There are many spatially-extended hybrid methods in which some species are represented using continuum models throughout the domain and others using discrete models in the same domain. These methods are popular when representing species which are inherently different in copy number throughout the domain. For example, small numbers of chemotaxing bacterial cells might be represented using an individual-based model, whereas the chemical signal to which they respond might be represented as a continuum. Since these models are not the primary focus of this review (rather we focus on models in which the same species is represented variably throughout the domain) we will give only a brief mention to some of these hybrid methods.

Cancerous tumour behaviour has frequently been represented using such hybrid methods. Anderson and Chaplain (1998) model angiogenesis – the directed growth of blood vessels towards the tumour. In order to do so they couple the macroscopic system of PDEs governing the growth of a tumour to a discrete model of blood vessel formation on a lattice. The discrete model is used in order to investigate how individual cells branch and undergo anastomosis and mitosis close to the tips of blood vessels which have sprouted. The authors also use a similar method to model the invasion of healthy tissue by a solid tumour (Anderson, 2005). Other examples of tumour growth hybrid methods include the use of cellular automata (Dormann and Deutsch, 2002; Gerlee and Anderson, 2007) and a method which models the environment as a continuum, while the tumour cells themselves are discrete and react the environment (Jeon et al., 2010). A similar idea has also been employed by Franz et al. (2013b), in which bacteria respond to a chemotactic signal. The signal is modelled by a continuum PDE, which the bacteria, modelled as individuals, can adapt and respond to.

7 Discussion and outlook

Within this review, we have explored the rich and diverse field of spatial hybrid methods, and illustrated how they can be utilised in order to probe previously intractable problems in the biological and physical sciences. Biological and physical phenomena exists at a variety of temporal, spatial and population scales (Dobramysl et al., 2015; Khan et al., 2011; Mort et al., 2016; Sherratt, 2005; Volpert and Petrovskii, 2009). Take, for example, the formation of calcium puffs at the endoplasmic reticulum (Dobramysl et al., 2015). Just before a calcium ion channel opens, the number of calcium ions is small. However, once the channel opens, the number of particles becomes orders of magnitude larger. Further away from the channels, particle numbers remain relatively small until diffusion disperses them. Even for a single phenomenon, populations can vary over orders of magnitude making traditional modelling approaches difficult. Novel modelling methods which span these scales in a computationally efficient manner may provide insights into these phenomena. This is precisely the purpose of many of the hybrid methods reviewed in this paper – they permit the representation of multiple scales within a system, allowing for efficient and accurate simulation. This review has focussed mostly on spatially-coupled hybrid methods for reaction-diffusion systems that allow space to be partitioned into subdomains in which different modelling paradigms are employed.

We covered couplings that broach four different spatial scales – the macro, meso and microscales, together with molecular dynamics. We have provided detailed summaries of illustrative examples for macroscopic-to-mesoscopic (PCM by Yates and Flegg (2015)), mesoscopic-to-microscopic (GCM by Flegg et al. (2015)) and macroscopic-to-microscopic (ARM by Smith and Yates (2017)) couplings, together with pseudocode for their implementation and demonstrations of worked examples, in order to facilitate the use of such hybrid methods. In addition, in the electronic supplementary material for this paper we provide working MATLAB code for each of the three methods. Schematics and descriptions of various

other methods provide an extensive yet non-exhaustive list of possible hybrid methods, which should be chosen depending on the application at hand, and the type of coupling desired.

Whilst not the focus of this review, there are other hybrid methods in which space is not modelled explicitly. Several hybrid methods concern the simulation of well-mixed chemical systems (Bentele and Eils, 2004; Burrage et al., 2004; Duncan et al., 2016; Hellander and Lötstedt, 2007; Salis and Kaznessis, 2005) while epidemiology (Bobashev et al., 2007) and stochastic reaction networks (Hepp et al., 2015) have also been investigated. We have also described several spatially-extended methods which used different types of hybridisation within section 6.

This review contains a summary of the current state of spatial hybrid methods. We now look to the future and directions in which the area will progress. Whilst much work has been completed within the field, there are still issues which are common to many of the methods. Chief amongst these is variation in hybrid methods that involve deterministic PDEs compared to the full solution simulated using a stochastic approach. Typically the deterministic nature of the continuum model results in damping of the variation in the stochastic subdomain in comparison to that of the fully stochastic method. Some authors have fixed this problem by incorporating an overlap region instead of an interface (Flekkøy et al., 2001; Franz et al., 2013a; Harrison and Yates, 2016). Within the overlap region, mass is simultaneously modelled using both representations. A second method for resolving the variance is to replace the PDE with an appropriate SPDE, a macroscopic model for which stochasticity is inherently incorporated. Provided the stochasticity is chosen in a consistent manner (consistent with the fully stochastic method), hybrid methods have been postulated for which the variance in the individual subdomain has been shown to match that of the fully stochastic model (Alexander et al., 2002).

As mentioned in Section 6, recently there has been work to couple microscopic descriptions to molecular dynamics. Erban (2014, 2016) has pioneered work in this area, providing methods which do just this. This type of method can be utilised in order to simulate biological phenomena at the molecular level, which even microscale Brownian motion may be unable to accurately capture.

There is a relative abundance of spatial hybrid methods (attested to by this review). Although we have presented a small number of papers which employ these methods in real physical and biological problems, there still remain very few practical applications of such methods. Whether this is due to the complexity of the hybrid methods in comparison to their single model counterparts or to the low profile of such methods, the challenge remains for the developers of such hybrid algorithms to realise the potential impact of their methods by applying them to real problems. We hope that this review has served the purpose of increasing the profile of hybrid methods, whilst simultaneously making them more accessible to the user.

Author contributions

CS and CY contributed equally to the production of this manuscript. CS performed the simulations and created the figures.

Acknowledgements

We would like to thank the anonymous reviewers for their constructive comments on the manuscript.

Data accessibility

The accompanying code has been uploaded as part of the supplementary material.

Funding

CS is supported by a scholarship from the EPSRC Centre for Doctoral Training in Statistical Applied Mathematics at Bath (SAMBa), under the project EP/L015684/1.

Competing interests

We have no competing interests.

References

T. Alarcón, H.M. Byrne, and P.K. Maini. A cellular automaton model for tumour growth in inhomogeneous environment. *J. Theor. Biol.*, 225(2):257–274, 2003.

F.J. Alexander, A.L. Garcia, and D.M. Tartakovsky. Algorithm refinement for stochastic partial differential equations: I. Linear diffusion. *J. Comput. Phys.*, 182(1):47–66, 2002.

F.J. Alexander, A.L. Garcia, and D.M. Tartakovsky. Algorithm refinement for stochastic partial differential equations: II. Correlated systems. *Journal of Computational Physics*, 207(2):769–787, 2005.

M.P. Allen and D.J. Tildesley. *Computer simulation of liquids*. Oxford university press, 2017.

A.R.A. Anderson. A hybrid mathematical model of solid tumour invasion: the importance of cell adhesion. *Math. Med. Biol.*, 22(2):163–186, 2005.

A.R.A. Anderson and M.A.J. Chaplain. Continuous and discrete mathematical models of tumor-induced angiogenesis. *Bull. Math. Biol.*, 60(5):857–899, 1998.

D.F. Anderson. A modified next reaction method for simulating chemical systems with time dependent propensities and delays. *J. Chem. Phys.*, 127(21):214107, 2007.

S.A. Andrews and D. Bray. Stochastic simulation of chemical reactions with spatial resolution and single molecule detail. *Phys. Biol.*, 1(3-4):137–151, 2004.

A. Auger, P. Chatelain, and P. Koumoutsakos. R-leaping: Accelerating the stochastic simulation algorithm by reaction leaps. *J. Chem. Phys.*, 125:084103, 2006.

M. Bentele and R. Eils. General stochastic hybrid method for the simulation of chemical reaction processes in cells. In V. Danos and V. Schachter, editors, *Computational Methods in Systems Biology 2004*. Springer, 2004.

M.J. Berger and P. Colella. Local adaptive mesh refinement for shock hydrodynamics. *J. Comput. Phys.*, 82(1):64–84, 1989.

D. Bernstein. Simulating mesoscopic reaction-diffusion systems using the gillespie algorithm. *Phys. Rev. E*, 71(4):041103, 2005.

G.V. Bobashev, D.M. Goedecke, F. Yu, and J.M. Epstein. A hybrid epidemic model: combining the advantages of agent-based and equation-based approaches. In *2007 Winter Simulation Conference*, pages 1532–1537. IEEE, 2007.

A.B. Bortz, M.H. Kalos, and J.L. Lebowitz. A new algorithm for Monte Carlo simulation of Ising spin systems. *J. Comput. Phys.*, 17(1):10–18, 1975.

S.C. Brenner and C. Carstensen. *Finite element methods*, 2004.

K. Burrage, T. Tian, and P. Burrage. A multi-scaled approach for simulating chemical reaction systems. *Prog. Biophys. Mol. Bio.*, 85(2-3):217–234, 2004.

Y. Cao, H. Li, and L. Petzold. Efficient formulation of the stochastic simulation algorithm for chemically reacting systems. *J. Chem. Phys.*, 121(9):4059–4067, 2004.

Y. Cao, D.T. Gillespie, and L.R. Petzold. The slow-scale stochastic simulation algorithm. *J. Chem. Phys.*, 122:014116, 2005.

Y. Cao, D.T. Gillespie, and L.R. Petzold. Efficient step size selection for the tau-leaping simulation method. *J. Chem. Phys.*, 124(4):044109, 2006.

K.-H. Chiam, C.M. Tan, V. Bhargava, and G. Rajagopal. Hybrid simulations of stochastic reaction-diffusion processes for modeling intracellular signaling pathways. *Phys. Rev. E*, 74(5):051910, 2006.

J.C. Dallon and H.G. Othmer. A discrete cell model with adaptive signalling for aggregation of Dictyostelium discoideum. *Phil. Trans. R. Soc. B*, 352(1351):391, 1997.

G. De Fabritiis, R. Delgado-Buscalioni, and P.V. Coveney. Multiscale modeling of liquids with molecular specificity. *Phys. Rev. Lett.*, 97(13):134501, 2006.

R. Delgado-Buscalioni, P.V. Coveney, G.D. Riley, and R.W. Ford. Hybrid molecular-continuum fluid models: implementation within a general coupling framework. *Phil. Trans. R. Soc. A*, 363(1833): 1975–1985, 2005a.

R. Delgado-Buscalioni, E.G. Flekkøy, and P.V. Coveney. Fluctuations and continuity in particle-continuum hybrid simulations of unsteady flows based on flux-exchange. *Europhys. Lett.*, 69(6):959, 2005b.

R. Delgado-Buscalioni, K. Kremer, and M. Praprotnik. Coupling atomistic and continuum hydrodynamics through a mesoscopic model: application to liquid water. *J. Chem. Phys.*, 131(24):244107, 2009.

U. Dobramysl, S. Rüdiger, and R. Erban. Particle-based multiscale modeling of intracellular calcium dynamics. *Multiscale. Model. Sim.*, 14(3):997–1016, 2015.

A. Donev, J.B. Bell, A.L. Garcia, and B.J. Alder. A hybrid particle-continuum method for hydrodynamics of complex fluids. *Multiscale. Model. Sim.*, 8(3):871–911, 2010.

S. Dormann and A. Deutsch. Modeling of self-organized avascular tumor growth with a hybrid cellular automaton. *In Silico Biol.*, 2(3):393–406, 2002.

B. Drawert, S. Engblom, and A. Hellander. URDME: a modular framework for stochastic simulation of reaction-transport processes in complex geometries. *BMC Syst. Biol.*, 6(1):76, 2012.

A. Duncan, R. Erban, and K. Zygalakis. Hybrid framework for the simulation of stochastic chemical kinetics. *J. Comput. Phys.*, 326:398–419, 2016.

D Dürr, S Goldstein, and JL Lebowitz. A mechanical model of brownian motion. *Commun. Math. Phys.*, 78(4):507–530, 1981.

J. Elf and M. Ehrenberg. Spontaneous separation of bi-stable biochemical systems into spatial domains of opposite phases. *Syst. Biol.*, 1(2):230–236, 2004.

R. Erban. *From individual to collective behavior in biological systems*. PhD thesis, University of Minnesota, 2004.

R. Erban. From molecular dynamics to brownian dynamics. *P. Roy. Soc. A-Math. Phy.*, 470(2167): 20140036, 2014.

R. Erban. Coupling all-atom molecular dynamics simulations of ions in water with brownian dynamics. *P. Roy. Soc. A-Math. Phy.*, 472(2186):20150556, 2016.

R. Erban and S.J. Chapman. Stochastic modelling of reaction–diffusion processes: algorithms for bimolecular reactions. *Phys. Biol.*, 6(4):1–18, 2009.

R. Erban, S.J. Chapman, and P. Maini. A practical guide to stochastic simulations of reaction-diffusion processes. *arXiv preprint arXiv:0704.1908*, 2007.

R. Erban, M.B. Flegg, and G.A. Papoian. Multiscale stochastic reaction–diffusion modeling: application to actin dynamics in filopodia. *Bull. Math. Biol.*, 76(4):799–818, 2014.

R. Eymard, T. Gallouët, and R. Herbin. Finite volume methods. *Handbook of numerical analysis*, 7: 713–1018, 2000.

M.B. Flegg, S.J. Chapman, and R. Erban. The two-regime method for optimizing stochastic reaction–diffusion simulations. *J. Roy. Soc. Interface*, 9(70):859–868, 2012.

M.B. Flegg, S. Rüdiger, and R. Erban. Diffusive spatio-temporal noise in a first-passage time model for intracellular calcium release. *J. Chem. Phys.*, 138(15):154103, 2013.

M.B. Flegg, S.J. Chapman, L. Zheng, and R. Erban. Analysis of the two-regime method on square meshes. *(SIAM) J. Sci. Comput.*, 36(3):B561–B588, 2014.

M.B. Flegg, S. Hellander, and R. Erban. Convergence of methods for coupling of microscopic and mesoscopic reaction-diffusion simulations. *J. Comput. Phys.*, 289(C):1–17, 2015.

E.G. Flekkøy and P.V. Coveney. From molecular dynamics to dissipative particle dynamics. *Phys. Rev. Lett.*, 83(9):1775, 1999.

E.G. Flekkøy, G. Wagner, and J. Feder. Hybrid model for combined particle and continuum dynamics. *Europhys. Lett.*, 52(3):271, 2000.

E.G. Flekkøy, J. Feder, and G. Wagner. Coupling particles and fields in a diffusive hybrid model. *Phys. Rev. E*, 64(6):066302, 2001.

B. Franz and R. Erban. *Hybrid modelling of individual movement and collective behaviour*. Springer, 2011.

B. Franz, M.B. Flegg, S.J. Chapman, and R. Erban. Multiscale reaction-diffusion algorithms: PDE-assisted Brownian dynamics. *SIAM J. Appl. Math.*, 73(3):1224–1247, 2013a.

B. Franz, C. Xue, K.J. Painter, and R. Erban. Travelling waves in hybrid chemotaxis models. *Bull. Math. Biol.*, 2013b.

A.L. Garcia, J.B. Bell, W.Y. Crutchfield, and B.J. Alder. Adaptive mesh and algorithm refinement using direct simulation Monte Carlo. *J. Comput. Phys.*, 154(1):134–155, 1999.

P. Gerlee and A.R.A. Anderson. An evolutionary hybrid cellular automaton model of solid tumour growth. *J. Theor. Biol.*, 246(4):583–603, 2007.

T. Geyer, C. Gorba, and V. Helms. Interfacing brownian dynamics simulations. *J. Chem. Phys.*, 120 (10):4573–4580, 2004.

M.A. Gibson and J. Bruck. Efficient exact stochastic simulation of chemical systems with many species and many channels. *J. Phys. Chem. A.*, 104(9):1876–1889, 2000.

D.T. Gillespie. Exact stochastic simulation of coupled chemical reactions. *J. Phys. Chem.*, 81(25): 2340–2361, 1977.

D.T. Gillespie. Approximate accelerated stochastic simulation of chemically reacting systems. *J. Chem. Phys.*, 115(4):1716–1733, 2001.

C. Gorba, T. Geyer, and V. Helms. Brownian dynamics simulations of simplified cytochrome c molecules in the presence of a charged surface. *J. Chem. Phys.*, 121(1):457–464, 2004.

Z. Guo, P.M.A. Sloot, and J.C. Tay. A hybrid agent-based approach for modeling microbiological systems. *J. Theor. Biol.*, 255(2):163–175, 2008.

N.G. Hadjiconstantinou. Hybrid atomistic-continuum formulations and the moving contact-line problem. *J. Comput. Phys.*, 154(2):245–265, 1999.

J.U. Harrison and C.A. Yates. A hybrid algorithm for coupling PDE and compartment-based dynamics. *J. Roy. Soc. Interface*, 13(122):20160335, 2016.

A. Hellander and P. Lötstedt. Hybrid method for the chemical master equation. *J. Comput. Phys.*, 227 (1):100–122, 2007.

A. Hellander, S. Hellander, and P. Lotstedt. Coupled mesoscopic and microscopic simulation of stochastic reaction-diffusion processes in mixed dimensions. *Multiscale. Model. Sim.*, 10(2):585–611, 2012.

B. Hepp, A. Gupta, and M. Khammash. Adaptive hybrid simulations for multiscale stochastic reaction networks. *The Journal of chemical physics*, 142(3):034118, 2015.

R. Holley. The motion of a heavy particle in an infinite one dimensional gas of hard spheres. *Probab. Theory and Relat. Fields*, 17(3):181–219, 1971.

S. Hoops, S. Sahle, R. Gauges, C. Lee, J. Pahle, N. Simus, M. Singhal, L. Xu, Pedro. Mendes, and U. Kummer. Copasi–a complex pathway simulator. *Bioinformatics*, 22(24):3067–3074, 2006.

S.A. Isaacson. Relationship between the reaction–diffusion master equation and particle tracking models. *J. Phys. A.-Math. Theor.*, 41(6):065003, 2008.

D.E. Jackson, S.J. Martin, M. Holcombe, and F.L.W. Ratnieks. Longevity and detection of persistent foraging trails in Pharaoh's ants, Monomorium pharaonis (L.). *Anim. Behav.*, 71(2):351–359, 2006.

J. Jeon, V. Quaranta, and P.T. Cummings. An off-lattice hybrid discrete-continuum model of tumor growth and invasion. *Biophysical journal*, 98(1):37–47, 2010.

M. Jeschke and A.M. Uhrmacher. Multi-resolution spatial simulation for molecular crowding. In *Simulation Conference, 2008. WSC 2008. Winter*, pages 1384–1392. IEEE, 2008.

S. Khan, Y. Zou, A. Amjad, A. Gardezi, C.L. Smith, C. Winters, and T.S. Reese. Sequestration of CaMKII in dendritic spines *in silico*. *J. Comput. Neurosci.*, 31(3):581–594, 2011.

T.R. Kiehl, R.M. Mattheyses, and M.K. Simmons. Hybrid simulation of cellular behavior. *Method. Biochem. Anal.*, 20(3):316–322, 2004.

M. Klann, A. Ganguly, and H. Koeppl. Hybrid spatial gillespie and particle tracking simulation. *Bioinformatics*, 28(18):i549–i555, 2012.

P. Koumoutsakos. Multiscale flow simulations using particles. *Annu. Rev. Fluid Mech.*, 37:457–487, 2005.

J.J. Landsberg and R.H. Waring. A generalised model of forest productivity using simplified concepts of radiation-use efficiency, carbon balance and partitioning. *Forest. Ecol. Manag.*, 95(3):209–228, 1997.

P. Langevin. Sur la théorie du mouvement brownien. *CR Acad. Sci. Paris*, 146(530-533):530, 1908.

H. Li and L. Petzold. Logarithmic direct method for discrete stochastic simulation of chemically reacting systems. Technical report, University of California Santa Barbara, 2006.

H. Li, Y. Cao, L.R. Petzold, and D.T. Gillespie. Algorithms and software for stochastic simulation of biochemical reacting systems. *Biotechnol. Progr.*, 24(1):56–61, 2008.

J. Lipková, K.C. Zygalakis, S.J. Chapman, and R. Erban. Analysis of brownian dynamics simulations of reversible bimolecular reactions. *SIAM J. Appl. Math.*, 71(3):714–730, 2011.

W.-C. Lo, L. Zheng, and Q. Nie. A hybrid continuous-discrete method for stochastic reaction–diffusion processes. *R. Soc. Open Sci.*, 3(9):160485, 2016.

J.M. McCollum, G.D. Peterson, C.D. Cox, M.L. Simpson, and N.F. Samatova. The sorting direct method for stochastic simulation of biochemical systems with varying reaction execution behavior. *Comput. Biol. Chem.*, 30(1):39–49, 2006.

K.M. Mohamed and A.A. Mohamad. A review of the development of hybrid atomistic–continuum methods for dense fluids. *Microfluid. Nanofluidics.*, 8(3):283–302, 2010.

E. Moro. Hybrid method for simulating front propagation in reaction-diffusion systems. *Phys. Rev. E*, 69(6):060101, 2004.

R.L. Mort, R.J.H. Ross, K.J. Hainey, O.J. Harrison, M.A. Keighren, G. Landini, R.E. Baker, K.J. Painter, I.J. Jackson, and C.A. Yates. Reconciling diverse mammalian pigmentation patterns with a fundamental mathematical model. *Nat. Commun.*, 7:10288, 2016.

K.W. Morton and D.F. Mayers. *Numerical Solution of Partial Differential Equations*. Cambridge University Press, 2005.

J.D. Murray. *Mathematical Biology II: Spatial Models and Biomedical Applications*, volume 2 of *Interdisciplinary Mathematics*. Springer, New York, 2003.

S.T. O'Connell and P.A. Thompson. Molecular dynamics–continuum hybrid computations: a tool for studying complex fluid flows. *Phys. Rev. E*, 52(6):R5792, 1995.

J.M. Osborne, A. Walter, S.K. Kershaw, G.R. Mirams, A.G. Fletcher, P. Pathmanathan, D. Gavaghan, O.E. Jensen, P.K. Maini, and H.M. Byrne. A hybrid approach to multi-scale modelling of cancer. *Phil. Trans. R. Soc. A*, 368(1930):5013–5028, 2010.

J. Pahle. Biochemical simulations: stochastic, approximate stochastic and hybrid approaches. *Brief. Bioinform.*, 10(1):53–64, 2009.

A.A. Patel, E.T. Gawlinski, S.K. Lemieux, and R.A. Gatenby. A cellular automaton model of early tumor growth and invasion: the effects of native tissue vascularity and increased anaerobic tumor metabolism. *J. Theor. Biol.*, 213(3):315–331, 2001.

M. Plapp and A. Karma. Multiscale random-walk algorithm for simulating interfacial pattern formation. *Phys. Rev. Lett.*, 84(8):1740, 2000.

N. Plattner, S. Doerr, G. De Fabritiis, and F. Noé. Complete protein–protein association kinetics in atomic detail revealed by molecular dynamics simulations and markov modelling. *Nature Chemistry*, 2017.

M. Praprotnik, L. Delle Site, and K. Kremer. Adaptive resolution molecular-dynamics simulation: Changing the degrees of freedom on the fly. *J. Chem. Phys.*, 123(22):224106, 2005.

B. Ribba, T. Alarcón, K. Marron, P.K. Maini, and Z. Agur. The use of hybrid cellular automaton models for improving cancer therapy. In *International Conference on Cellular Automata*, pages 444–453. Springer, 2004.

E. Roberts, J.E. Stone, and Z. Luthey-Schulten. Lattice microbes: High-performance stochastic simulation method for the reaction-diffusion master equation. *J. Comput. Chem.*, 34(3):245–255, 2013.

M. Robinson, M. Flegg, and R. Erban. Adaptive two-regime method: application to front propagation. *J. Chem. Phys.*, 140(12):124109, 2014.

D. Rossinelli, B. Bayati, and P. Koumoutsakos. Accelerated stochastic and hybrid methods for spatial simulations of reaction–diffusion systems. *Chem. Phys. Lett.*, 451(1):136–140, 2008.

H. Salis and Y. Kaznessis. Accurate hybrid stochastic simulation of a system of coupled chemical or biochemical reactions. *J. Chem. Phys.*, 122:054103:1–13, 2005.

T.P. Schulze, P. Smereka, and W. E. Coupling kinetic monte-carlo and continuum models with application to epitaxial growth. *J. Comput. Phys.*, 189(1):197–211, 2003.

V.B. Shenoy, R. Miller, E.B. Tadmor, D. Rodney, R. Phillips, and M. Ortiz. An adaptive finite element approach to atomic-scale mechanics – the quasicontinuum method. *J. Mech. Phys. Solids.*, 47(3): 611–642, 1999.

J.A. Sherratt. An analysis of vegetation stripe formation in semi-arid landscapes. *J. Math. Biol.*, 51(2): 183–197, 2005.

K. Smallbone, R.A. Gatenby, R.J. Gillies, P.K. Maini, and D.J. Gavaghan. Metabolic changes during carcinogenesis: potential impact on invasiveness. *J. Theor. Biol.*, 244(4):703–713, 2007.

C.A. Smith and C.A. Yates. The auxiliary region method: A hybrid method for coupling a PDE to Brownian-based dynamics for reaction-diffusion systems. *https://arxiv.org/abs/1708.04457*, 2017.

G.D. Smith. *Numerical solution of partial differential equations: finite difference methods*. Oxford university press, 1985.

M. Smoluchowski. Versuch einer mathematischen theorie der koagulationskinetik kolloider lösungen. *Z. Phys. Chem.*, 92(129-168):9, 1917.

F. Spill, P. Guerrero, T. Alarcon, P.K. Maini, and H. Byrne. Hybrid approaches for multiple-species stochastic reaction–diffusion models. *J. Comput. Phys.*, 299:429–445, 2015.

J. Stiles and T. Bartol. *Monte Carlo Methods for Simulating Realistic Synaptic Microphysiology Using MCell, in, "Computational Neuroscience, Realsitic Modelling for Experimentalists"*, chapter 4, pages 87–127. CRC Press, 2001.

E.B. Tadmor, M. Ortiz, and R. Phillips. Quasicontinuum analysis of defects in solids. *Philos. Mag. A.*, 73(6):1529–1563, 1996.

M. Tomita, K. Hashimoto, K. Takahashi, T.S. Shimizu, Y. Matsuzaki, F. Miyoshi, K. Saito, S. Tanida, K. Yugi, J.C. Venter, and C.A. Hutchinson. E-cell: software environment for whole-cell simulation. *Bioinformatics*, 15(1):72–84, 1999.

J.S. van Zon and P.R. ten Wolde. Green's-function reaction dynamics: a particle-based approach for simulating biochemical networks in time and space. *J. Chem. Phys.*, 123(23):234910, 2005.

V. Volpert and S. Petrovskii. Reaction–diffusion waves in biology. *Phys. Life Rev.*, 6(4):267–310, 2009.

G. Wagner and E.G. Flekkøy. Hybrid computations with flux exchange. *Phil. Trans. R. Soc. A*, 362: 1655–1666, 2004.

G. Wagner, E. Flekkøy, J. Feder, and T. Jøssang. Coupling molecular dynamics and continuum dynamics. *Computer physics communications*, 147(1):670–673, 2002.

S.A. Williams, J.B. Bell, and A.L. Garcia. Algorithm refinement for fluctuating hydrodynamics. *Multiscale Model. Simul.*, 6(4):1256–1280, 2008.

S. Wils and E. De Schutter. STEPS: modeling and simulating complex reaction-diffusion systems with Python. *Front. Neuroinf.*, 3, 2009.

D.C. Wylie, Y. Hori, A.R. Dinner, and A.K. Chakraborty. A hybrid deterministic-stochastic algorithm for modeling cell signaling dynamics in spatially inhomogeneous environments and under the influence of external fields. *J. Phys. Chem. B*, 110(25):12749–12765, 2006.

C. Xue and H.G. Othmer. Multiscale models of taxis-driven patterning in bacterial populations. *SIAM J. Appl. Math.*, 70(1):133, 2009.

C.A. Yates and M.B. Flegg. The pseudo-compartment method for coupling partial differential equation and compartment-based models of diffusion. *J. Roy. Soc. Interface*, 12(106):20150141, 2015.

C.A. Yates and G. Klingbeil. Recycling random numbers in the stochastic simulation algorithm. *J. Chem. Phys.*, 138(9):094103, 2013.

2.2 Post-publication literature review

The paper component of this chapter contains a review of the literature up to the beginning of 2018. We begin this concluding part of the chapter by investigating some of the further papers that have either been written since the date of publication of the review paper, or have come to our attention in the context of projects undertaken since 2018.

We begin with de la Cruz et al. (2017), who present a spatially extended hybrid method that couples a macroscopic to a mesoscopic representation of an age-dependent population of cells applied to tumour growth. The method builds upon the work of Spill et al. (2015) (described in the main review paper that comprises this chapter) with additional structure and a mesoscopic representation which is also multiscale. Cells have an associated age, and are compartmentalised with the ability to undergo three processes: birth, death and diffusion. Birth occurs with a rate which depends on the age of the cell, with older cells more likely to produce new cells. When a birth event occurs, the cell which undergoes the birth process is removed, and two new cells with age zero are created. If a cell is chosen to die (with a constant rate), it is simply removed from the compartment. Finally, cells are able to jump between compartments using the usual diffusive jumping rates. The age dependent birth rate depends on the solution of a PDE which determines the concentration of oxygen, which in turn is influenced by the state of the compartments.

The stochastic method outlined above is then coupled to the mean-field approximation of the process with averaged age profile through the interfacial method of Spill et al. (2015). As a reminder, an interfacial compartment acts as an intermediary between the two modelling regimes using both the macroscopic and mesoscopic representations, with flux towards the macroscale being governed by that regime, and the jumps from this compartment to the mesoscale completed using the regular jump rate. All reactions within this interfacial compartment are done via the mesoscale (see Figure 3 of the paper in this chapter for a schematic of the method suggested by Spill et al. (2015)). The main difference between the method of de la Cruz et al. (2017) and that of Spill et al. (2015) is the way in which the authors reconcile the age structure when converting between the two regimes. When a cell jumps from the macroscopic regime to the mesoscopic subdomain, it needs to be assigned an age. This is done by sampling from the age-structure distribution at equilibrium. When a cell is removed from this interfacial compartment, an age is sampled from the age-structured stationary distribution at the

interface and a cell with that age is removed from the compartment. The method of de la Cruz et al. (2017) also employs an adaptive interface, which works in the same way as for Spill et al. (2015). Again, if ages need to be assigned, the age-structured equilibrium distribution is employed for each cell.

Next, we look at two macroscopic-to-mesoscopic methods, presented by Kang and Erban (2019). These spatially extended hybrid methods couple a stochastic partial differential equation (SPDE) to the mesoscopic representation. Each of the two schemes rely on using the mesoscopic approach in order for mass to cross between the two subdomains, analogously to the pseudo-compartment method (Yates and Flegg, 2015), and as such, continuous mass, as modelled by the SPDE, is not able to flow across the interface, requiring a zero-flux boundary.

In the first scheme, particles jump across the interface with the usual mesoscopic jumping rate between the first of the compartments (adjacent to the interface) and the so-called "handshaking region" in the SPDE subdomain (analogous to the pseudo-compartment of the PCM), which is of the same width as the compartments. The handshaking region is further split into an integer number of SPDE mesh points. When a particle is deemed to have jumped from the mesoscopic to the macroscopic subdomain, a particle is removed from compartment 1, and is randomly added to one of the SPDE nodes within the handshaking region.

The second of the two methods uses two unknown parameters that need to be calculated in order to modify the jump rates. On this occasion, particles can only jump between the first compartment and the final SPDE node, each adjacent to the interface. This requires an update to the jump parameters, which are calculated in the paper as the ratio of the length of the region being jumped from, to the average of the lengths of the beginning and destination locations. These rates are then multiplied by the usual jump rates in order to obtain jump rates between two different sized regions.

Finally, we investigate a method proposed by Kostré et al. (2020), which has some properties that make it similar to the auxiliary region method. The authors introduce a macroscopic-to-microscopic hybrid method, with a specific emphasis on the macroscale representing a particle reservoir, in a similar way to the paper of Gorba et al. (2004) which has a constant density heat bath. However, the key difference here is that the reservoir continues to undergo reactions and diffusion via the analytical or numerical solution of an appropriate PDE. The paper describes a coupling in two dimensions, where small squares of a pre-prescribed size, called boundary cells, are set up on either side of an interface.

These boundary cells allow us to couple the two representations through three processes: "injection" which allows reservoir particles to enter the particle domain; "reaction" for particles on either side of the interface to interact with one another; and "diffusion" in order to update all particles in the particle-based subdomain, including those that have been added in either of the two previous steps. Firstly, virtual particles are initialised inside the PDE boundary cells by using the average concentration across the cell, and then converting this to a number of particles. Each particle is afforded a rate with which it may jump into the neighbouring boundary cell in the particle-based subdomain. Any fractional virtual particles jump across the interface with the corresponding fraction of the rate. All particles that successfully jump across the interface are initialised by sampling a uniform position in the neighbouring boundary cell. Next, particles are able to react with the virtual particles in the PDE subdomain using an individual-based approach. Any virtual particles which are fractional are converted to a particle or removed with probability equal to their fraction, and then all virtual particles are uniformly placed within the boundary cell. Any second-order reactions that are deemed to take place between particles in the boundary cells are completed, and a new particle is placed at the central point of the two reactant particles. If this central point resides within the particle subdomain it is kept, otherwise it is removed. Finally, particles (including newly added ones, but not the virtual ones) are able to diffuse. If they diffuse into the PDE subdomain they are deleted.

The papers presented in this final section are, to the best of our knowledge, all of the spatially extended hybrid methods that have been published since the original review paper was written.

2.3 Conclusions

This chapter contains a comprehensive review of spatially extended hybrid methods which is intended to be used as a reference guide and manual for creating or employing hybrid methods. It contains several illustrative examples comprising algorithms, sample results and accompanying code, together with descriptions of several other methods of similar types, many of which are accompanied by cartoon schematics to help explain them.

This paper has the potential to have a large impact on the mathematical biological community. Up until the date of publication, there were very few papers using spatially extended (or other) hybrid methods to represent biologically relevant examples. This

could be for a number of reasons, but is perhaps in part due to the fact that the wider biological and mathematical communities are not aware of these methods. Part of the motivation behind this paper is to bring the idea of hybrid methods to the forefront for those that may not know about them. A second potential reason why hybrid methods are not currently employed as frequently as they could be is the idea that they are difficult to implement. This is where the illustrative examples will be of great importance. These show firstly that the hybrid methods are not necessarily complicated to simulated, and secondly, given that there is knowledge of the individual paradigms, that hybrid methods are an extension of those and can be easily implemented.

In the broader context of the thesis, the next chapter contains the paper which fully describes the auxiliary region method, the illustrative example for the macroscale-to-microscale hybrid methods. Looking further on, Chapter 6 contains each of the three illustrative examples extended onto a uniformly growing domain, something which hasn't been done using the spatially extended hybrid methods approach.

Chapter 3

The auxiliary region method: a hybrid method for coupling PDE- and Brownian-based dynamics for reaction-diffusion systems

This chapter contains a paper published in 2018 in the journal *Royal Society: Open Science*, co-authored by myself and Kit Yates, which develops a macroscopic-to-microscopic spatially extended hybrid method, called the auxiliary region method (ARM). This paper was written primarily in order to create a compartment-based coupling method, similar to the pseudo-compartment (Yates and Flegg, 2015) and ghost cell methods (Flegg et al., 2015), allowing for a unified approach. These types of couplings utilise additional compartments in order to allow mass to pass across an interface into the neighbouring region. Moreover, while researching for the review paper in Chapter 2, we discovered that one of the most widely employed macro-to-micro methods at the time, that of Franz et al. (2013), is very sensitive to the parameter set that is used.

The method presented by (Franz et al., 2013) often fails to equilibrate to the correct steady state even under the simplest of initial conditions. In particular, the problems seem to arise when the time-step used to progress the numerical solution is either too big or too small. If the time-step is too small, we observe that the expected mass that crosses the boundary according to a theoretical diffusion kernel is too large in comparison to the hybrid method. In contrast, if the time-step is too large, the

probability of initiating a new particle becomes greater than one. As a result of this, the method described in this chapter's paper is also a valuable tool for the biological and mathematical communities who are attempting to model problems with significant differences in particle numbers across the spatial domain, or requiring individual-level detail only in specific regions of space, such as when modelling calcium ion channels (Dobramysl et al., 2015).

In order to test whether our hybrid method is accurate at the mean-field level, we need to compare it to a "ground truth". This is defined to be the Fokker-Planck equation associated with a diffusive process with zero-flux boundary conditions, multiplied by the (fixed) number of particles in the system to convert from a probability to particle numbers, for test problems 1 and 2. For test problem 3, we find the diffusive limit of the RDME for the mean-field dynamics of the underlying mesoscopic process and use this as the ground truth. Finally, for test problem 4, we compare the ARM to 1000 independent repeats of the purely Brownian-based simulation across the entire domain.

We note here that the mean-field approximation is not the only way to assess the accuracy of our methods. For example, for the RDME given in Section 4.4.1 of the paper, we are able to calculate a PDE for the evolution of the moment generating function, and to use this to solve the RDME exactly. This allows an alternative approach to assess the accuracy of our method. We also employ a full implementation of the microscale simulation as a potential method of assessing our hybrid method (as in Section 4.4 of the paper below).

It should also be mentioned that the mean number of particles in this final example under a Brownian dynamics framework have different properties compared to the mean-field PDE due to the coupling between the mean and variance at steady state (Smith et al., 2016). In this paper, the authors demonstrate that there is a relationship between the number of particles in the system and the diffusion coefficient in the Brownian dynamics simulation, which is not present in the PDE system, demonstrating the complexities regarding second-order reactions and the possible need for PDEs for higher order moments when in such situations. Hybrid methods in which the macroscale model is solved using multiple moments (such as mean-field and variance) may help alleviate some of the predominant issues which exist in the hybrid methods literature, such as variance correction (see Chapter 7 for more information).

In the wider context of the thesis, this method is one of the three methods that will later be adapted to work on a growing domain in Chapter 6. Section 3 of this paper, where we describe the ARM in detail, is therefore of importance to Chapter 6. Some

of the test problems introduced in Section 4 will also be utilised in subsequent chapters of the thesis, in particular the examples in Section 4.1 (maintaining equilibrium) and 4.3 (the formation of a morphogen gradient).

3.1 Outline of the paper

In Section 1 we give a general overview of reaction-diffusion systems and their applications, and describe hybrid methods and the required individual modelling paradigms. Section 2 looks at replicating the results of Franz et al. (2013) and demonstrating the discrepancy between the hybrid method and the known analytical solution. The next three sections are concerned with the development, specification and subsequent testing of the ARM. Firstly, in section 3 we describe the method in detail, including a comprehensive algorithm for its implementation in Section 3.7, while in Section 4 we apply the ARM to four test problems of varying difficulties, each designed to test various aspects of the coupling. Section 5 investigates the accuracy of the method in more detail and also demonstrates that it is insensitive to different parameter regimes. Finally, there is a short discussion of the method and its potential impact in Section 6.

Appendix 6B: Statement of Authorship

This declaration concerns the article entitled:			
The auxiliary region method: a hybrid for coupling PDE- and Brownian-based dynamics for reaction-diffusion systems			
Publication status (tick one) **Draft manuscript** ☐ **Submitted** ☐ **In review** ☐ **Accepted** ☐ **Published** ●			
Publication details (reference)	Journal – Royal Society: Open Science, 5 (8), 180920 Authors – Cameron A. Smith, Christian A. Yates		
Copyright status (tick the appropriate statement) I hold the copyright for this material ● Copyright is retained by the publisher, but I have been given permission to replicate the material here ☐			
Candidate's contribution to the paper (provide details, and also indicate as a percentage)	All calculations have been performed by the author of this thesis (100%). All numerical computations and simulations have been completed by the author of this thesis (100%). All authors contributed equally to the presentation of the content (50%).		
Statement from Candidate	This paper reports on original research I conducted during the period of my Higher Degree by Research candidature.		
Signed		**Date**	04/06/2021

The auxiliary region method: A hybrid method for coupling PDE- and Brownian-based dynamics for reaction-diffusion systems

Cameron A. Smith[1,*], Christian A. Yates[1,**]
[1]**Centre for Mathematical Biology, Department of Mathematical Sciences, University of Bath, Claverton Down, Bath, BA2 7AY, United Kingdom**
∗ **E-mail: c.smith3@bath.ac.uk**
∗∗ **E-mail: c.yates@bath.ac.uk**

Key words: hybrid modelling, stochastic reaction-diffusion, multiscale modelling, auxiliary region, partial differential equation, Brownian dynamics

Abstract

Reaction-diffusion systems are used to represent many biological and physical phenomena. They model the random motion of particles (diffusion) and interactions between them (reactions). Such systems can be modelled at multiple scales with varying degrees of accuracy and computational efficiency. When representing genuinely multiscale phenomena, fine-scale models can be prohibitively expensive, whereas coarser models, although cheaper, often lack sufficient detail to accurately represent the phenomenon at hand. Spatial hybrid methods couple two or more of these representations in order to improve efficiency without compromising accuracy.

In this paper, we present a novel spatial hybrid method, which we call the auxiliary region method (ARM), which couples PDE and Brownian-based representations of reaction-diffusion systems. Numerical PDE solutions on one side of an interface are coupled to Brownian-based dynamics on the other side using compartment-based "auxiliary regions". We demonstrate that the hybrid method is able to simulate reaction-diffusion dynamics for a number of different test problems with high accuracy. Further, we undertake error analysis on the ARM which demonstrates that it is robust to changes in the free parameters in the model, where previous coupling algorithms are not. In particular, we envisage that the method will be applicable for a wide range of spatial multi-scales problems including, filopodial dynamics, intracellular signalling, embryogenesis and travelling wave phenomena.

1 Introduction

Reaction-diffusion models are important mathematical tools that are used to represent and understand complex biological and physical behaviours. They model the random movement of the particles (diffusion) and the interactions between particles (reactions), giving them a wide array of applications across multiple spatial scales. These applications range from the large-scale representation of striped vegetation patterns in semi-arid landscapes [41] and the spread of epidemics [47] to smaller-scale studies of pattern formation during embryogenesis [45, 36] and, at even smaller scales, to the study of actin dynamics inside a cell's filopodia [14] and intracellular dynamics [32, 2, 50].

Reaction-diffusion models can be specified at different levels of detail depending on the temporal, spatial and concentration scales involved in the application (see Table 1). At the finest scale that we will consider are microscopic dynamics. These models and methods (which include Brownian motion for purely diffusive systems and Smoluchowski dynamics [3, 44] or Green's function reaction dynamics (GFRD) [46] for reaction-diffusion systems) are amongst the more detailed representations of such systems, but consequently are relatively computationally expensive[1]. They require not only the knowledge

[1]Throughout this paper, regardless of whether we have interactions between particles or not, we shall refer to models at this microscopic scale as "Brownian dynamics".

of the location of all particles at all times, but in the case of second- and higher-order reactions, the pairwise distances between particles, which requires large memory, and are expensive to calculate for many time-steps. In the case of diffusion-limited reactions, time-steps must be taken to be extremely small to ensure that reactive particles do not jump past each other and that the attendant reaction events are not missed. All update steps also require the production of a normally distributed random number for each co-ordinate of each particle which can be computationally expensive depending on the reaction system that is being modelled. However, some of these expensive steps can be accelerated by considering event-driven algorithms or employing approximate algorithms with longer time-steps. GFRD is an event-driven algorithm differs from the standard method for simulating Brownian motion. It uses a maximum time-step so that only single particles, or pairs of particles, need to be considered. It then utilises the exact solution to the Smoluchowski equation in order to combine movement of, and interactions between, particles. If particles are far apart, the event-based time-steps are large. Smoldyn uses relatively long time-steps, and accounts for the error that this causes (due to possible reactant pairs passing by one another without the possibility of reacting) by making the effective particle sizes larger. Micro-scale modelling is particularly useful when fine scale detail is required, for example, when considering the binding of particles to receptors [8, 12, 38]. An even finer scale representation, in which atomistic dynamics can be represented, is available, if required. Typically, modelling at this scale is known as molecular dynamics, and we direct the reader to Holley [30] and Dürr et al. [10] for more information about modelling at this scale.

At a coarser scale we have compartment-based or mesoscopic models. Like the fine-scale, microscopic models, these also account for stochastic variation, however particles are now considered to belong to compartments rather than having their exact locations tracked. Particles can either react with one another within a compartment, or can jump between adjacent compartments with given rates, simulating diffusion. Compartment-based models can be simulated using either exact but computationally expensive [24, 23, 11] or inexact but computationally cheaper [25] stochastic simulation algorithms (SSAs). The exact methods (so-called because they produce sample paths consistent with the associated chemical master equation) effectively assign exponential waiting times to every possible event (diffusive jump or reaction) and then choose the event with the shortest waiting time to enact. In general, they are faster than the microscopic methods, since pairwise reaction distances do not need to be calculated for bimolecular reactions and individual particle identities are not tracked, but are less accurate, since they only record a particle's location up to the accuracy of the compartment size, and generally particles are only allowed to react with others in the same compartment [31].

Finally, at the coarsest scale lie continuum or macroscopic models. The most commonly employed macroscopic models for reaction-diffusion systems comprise partial differential equations (PDEs)[2]. These methods are generally only valid for high particle numbers. The stochastic variations, which are considered small enough to be neglected at high copy numbers, play a pivotal role in the dynamics at low copy numbers, leading the PDE solutions to diverge from the true underlying dynamics. There is a wealth of well established numerical methods that can quickly simulate an approximate solution to a PDE. These include finite-difference methods, finite-volume methods and finite-element methods (see for example, [43, 37, 15, 5]).

Often though, important biological and physical phenomena are genuinely multiscale [34, 4, 26, 39]. In spatial reaction-diffusion systems, concentration may vary over orders of magnitude. In regions of low concentration it is often important to employ detailed individual-based models in order to correctly represent the dynamics. If these models were to be employed indiscriminately throughout the domain, however, the regions of high concentration, in which there are many individual particles to be evolved, might render the system computationally intractable. In these regions it might be acceptable to employ a coarser and less computationally expensive model. A canonical example of this phenomenon is the

[2]However, with the increasing awareness of the importance of randomness, stochastic partial differential equations (SPDEs) are also becoming popular macroscopic representations.

Scale	Advantages	Disadvantages
Micro	Most accurate representation.	Slow to compute reactions.
	Can be used for low copy numbers.	Impractical for large numbers of particles.
Meso	Fast for low particle numbers.	Can be slow for large copy numbers.
	Represents individual-level behaviour.	Does not retain precise location or particle identity.
Macro	Fast to compute solutions.	Inaccurate for low copy numbers.
	Suitable for high copy numbers.	Mean-field models diverge from individual
	Often amenable to analytical solutions.	dynamics for higher-order reactions.

Table 1. A comparison of the advantages and disadvantages of the three most prominent scales at which reaction-diffusion processes are modelled.

stochastic Fisher wave [7, 6]. The wave speed is determined by the stochastic activity at the pulled front, so it is important to employ an accurate individual-based representation of the dynamics in this region. Conversely, behind the wave front, the detailed dynamics are of little importance. It is possible, therefore, to employ a coarser, cheaper representation of the dynamics in this region.

Spatially coupled hybrid methods have been developed for precisely this purpose: to simulate spatially inhomogeneous domains both accurately and efficiently. In general, such methods are designed to accelerate expensive computations whilst maintaining reasonable levels of accuracy. The majority of spatially coupled hybrid methods divide the computational domain into distinct regions using interfaces. The dynamics of adjacent regions are represented using different methods. Regions in which detailed representations of the dynamics are required for accuracy are simulated using a fine-scale method, whereas regions in which less detail is required are modelled with a coarser, less computationally expensive method. There can be two reasons for this. The first is in order to resolve a particular region of the spatial domain in more detail, such as when looking at the behaviour of ions around gated channels [8], or when building a model for the energy in a liquid crystal [40]. Both of these examples have a prohibitively slow but accurate model that is required in certain regions of space, but which is too computationally expensive if used everywhere. The second reason is to simply segregate a region of the domain in which there are very few particle numbers. In these regions a coarse method (for example a continuum model) may be too inaccurate.

There exist hybrid methods that couple each of the different scales described above to one another (and indeed many more, see Smith and Yates [42] for a comprehensive review of such methods). Macroscopic-to-mesoscopic methods have been proposed which employ averaged fluxes in order to calculate appropriate boundary conditions for each regime at the interface(s) [48, 35, 28], as well as using an extra compartment within the macroscopic region [49]. Mesoscopic-to-microscopic methods, which also employ extra compartments, this time in the microscopic regime, have been developed [19], and a class of methods using adapted rates of diffusion from the mesoscopic to the microscopic domains have been proposed and successfully applied to represent biological processes [17, 39, 18, 8, 14]. There are fewer macroscopic-to-microscopic hybrid methods in the literature. Macro-to-micro methods that allow mass to flow over the interface in both directions in order to initialise particles [21] or that average solutions on either side of the interface to find a flux [1] can be found in the literature. For a more detailed review of spatially extended hybrid methods, see [42].

Two of the above-mentioned hybrid methods are of particular relevance for the purposes of this paper. The pseudo-compartment method, presented by Yates and Flegg [49], is a macroscopic-to-mesoscopic (specifically PDE-to-compartment) method in which the coupling is achieved using an extra compartment, known as the "pseudo-compartment", adjacent to the interface within the macroscopic domain. In this compartment, mass is represented using both the PDE solution and the compartment-based method (with

particle numbers found by direct integration of the PDE over this region). Particles are then allowed to cross the interface in both directions using the compartment-based method. We give a schematic representation of this method in Figure 1 (a).

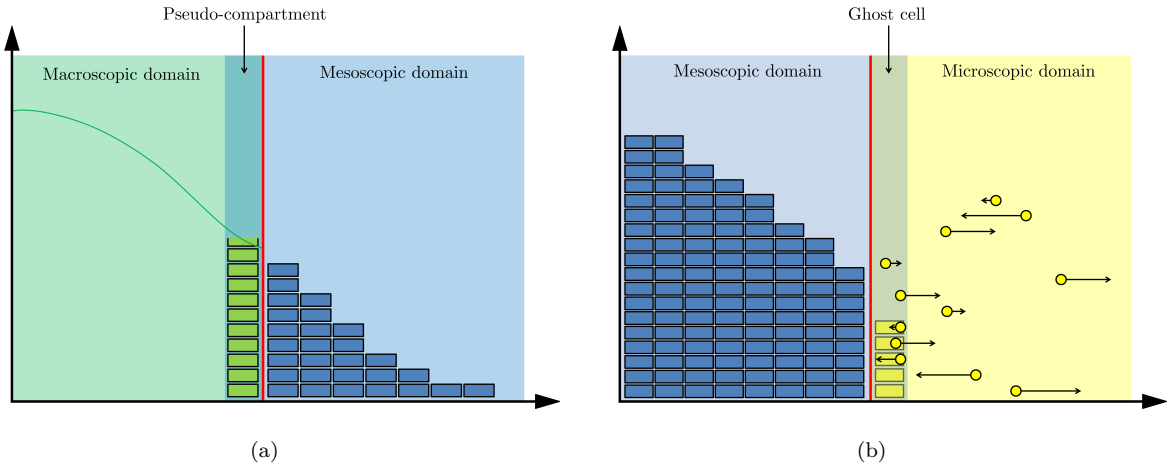


Figure 1. Schematics for (a) the pseudo-compartment method and the (b) the ghost cell method. The green line represents the PDE solution, the blue boxes the number of particles in the mesoscopic region of the respective hybrid methods, and the yellow dots denote the Brownian particles. These particles are shown with a volume, but in the simulations do not have a mass or volume. In the scenario illustrated, the particles reside on the one-dimensional line, but have been illustrated in the plane in order to show the directions and magnitudes of their next movement clearly (black arrows). The green boxes in (a) denote the number of particles in the pseudo-compartment, and similarly, the yellow boxes in (b) are the number of particles in the ghost cell, with each box representing a single particle. In each case, the red line denotes the point interface between the two regimes.

The ghost cell method proposed by Flegg et al. [19] is a mesoscopic-to-microscopic method which uses an extra compartment in the microscopic domain. The number of particles in this "ghost cell" is simply the number of Brownian particles which reside in this region. Again, particles are allowed to jump across the interface using the compartment-based mesoscopic method. A schematic representation of the method is given in Figure 1 (b).

In this paper, we employ the two methods described above (see Figure 1) in order to couple a macroscopic PDE description for reaction-diffusion systems to a corresponding microscopic Brownian dynamics representation through the use of "auxiliary regions". These regions are compartments, which lie either side of the interface, and allow mass to pass between the two regimes via a mesoscopic jump process (see Figure 4 for a schematic representation). Within the auxiliary regions, mass is simultaneously represented using both the description for the region in which they reside (i.e. PDE or Brownian) and the mesoscopic description. Changes (i.e. reactions or diffusion events) implemented under one modelling paradigm (e.g. the compartment-based representation of the auxiliary region) are simultaneously implemented in the other (e.g. the PDE or Brownian representations in these regions). The interface, which divides the two modelling paradigms, can either be static, in which case it remains in its initial position, or adaptive, in which case it moves with the density profile in order to ensure that regions of space with few particles are simulated using the finest scale. Through a series of test cases, we demonstrate our algorithm to be more accurate and more robust to model parameters than previous PDE-to-Brownian coupling algorithms.

The paper is organised as follows. In Section 2, a previous attempt at hybridising a Brownian dynamics

model to its corresponding mean-field PDE description is evaluated in more detail [21]. A description of our novel auxiliary region method (ARM) is presented in Section 3 alongside the relevant justifications and pseudocode. Numerical results, verifying the accuracy of our hybrid method, are presented in Section 4. Numerical error analysis is conducted in Section 5, where we also discuss restrictions on the model parameters for the effective functioning of the coupling algorithm. We conclude with a discussion of the effectiveness of our new hybrid method and suggest avenues for further exploration in Section 6.

2 An existing PDE-to-Brownian coupling

In this section we summarise the pioneering work of Franz et al. [21], who were among the first to couple PDE and Brownian dynamics representations of reaction-diffusion. By replicating their results, we demonstrate that their "PDE-assisted Brownian dynamics" algorithm is not robust to simulation parameter choice, even for simple diffusive processes. This motivates the need for a more robust coupling method, which we provide in the form of the ARM in Section 3.

2.1 PDE-assisted Brownian dynamics

Hybrid methods that couple the PDE description of a reaction-diffusion system to its corresponding Brownian dynamics representation have been relatively poorly investigated in comparison to PDE-to-compartment-based and compartment-based-to-Brownian couplings. In part, this is a result of the fact that such hybrid algorithms neglect meso-scale representations of particle dynamics, meaning that they must bridge a greater scale separation than either of the other two hybrid paradigms. Mainly though, the absence of many examples of PDE-to-Brownian hybrid methods is due to the inherent difficulty when converting PDE mass to individual particles (and vice-versa) when coupling Brownian dynamics models to continuum PDE representations. Below, we describe two algorithms proposed by Franz et al. [21], but focus on the first, a method with an interfacial coupling. We choose to focus on this coupling because our ARM coupling method, described in Section 3, also utilises an interface.

Franz et al. [21] present two related algorithms. In the first, the non-overlapping PDE and Brownian domains are separated by an interface (see Figure 2). Both PDE and Brownian representations are updated using a time-driven algorithm, with the PDE time-step much smaller than the Brownian time-step. The discretised PDE is evolved (until the time reaches the next Brownian time-step) using a centred finite-difference scheme with implicit Euler time-stepping, and PDE mass is allowed to cross the interface between the two regimes. Provided that the Brownian time-step is sufficiently small, the amount of mass that crosses the interface between Brownian time-steps gives the probability that a new particle is placed within the Brownian domain. A uniformly distributed random number is used to determine whether a particle is initialised in the Brownian regime or not. If it is, this particle's position is randomly initialised according to the normalised density profile of the PDE mass that crossed the interface in the previous Brownian time-step. If a Brownian particle crosses into the PDE domain, a particle's worth of mass is added to the PDE solution at its new location as a δ-function and the individual particle is removed. We have illustrated this method schematically in Figure 2.

Franz et al. [21] found the variance in particle numbers in the Brownian region of the hybrid domain to be altered in comparison to the variance that would be expected in a fully Brownian simulation. In order to counteract this problem, they introduced a second algorithm, in which an overlap region replaces the interface. Within the overlap region, mass can be simulated as either Brownian particles or as part of the PDE. The coupling works in the same way as in the interfacing algorithm, however the Brownian particles are subsumed into the PDE only once they have crossed the boundary of the overlap region closest to the fully-PDE domain. Similarly, PDE mass can only be converted to Brownian particles once it has flowed over the overlap boundary adjacent to the fully-Brownian domain.

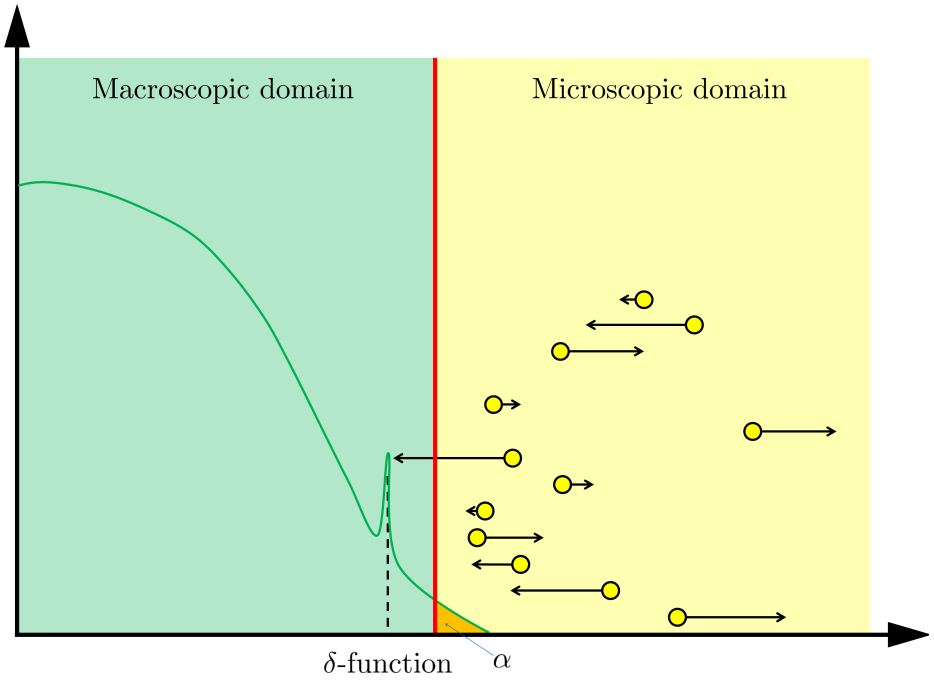


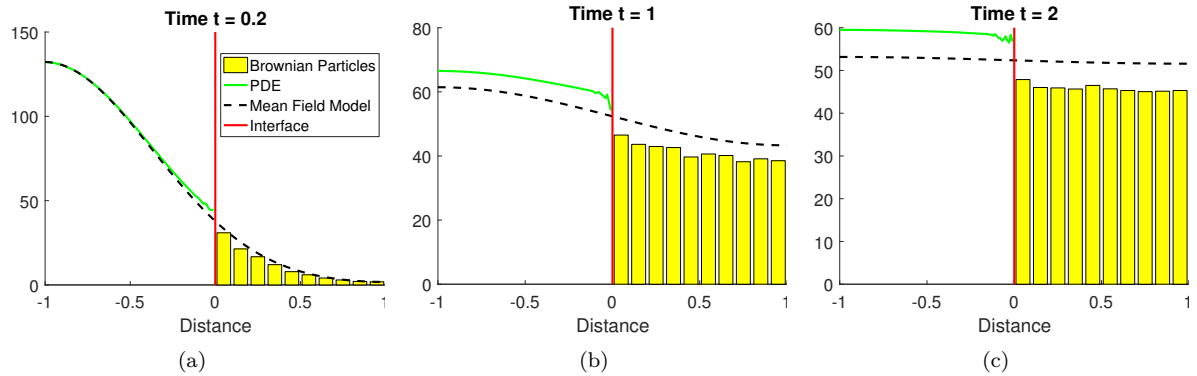
Figure 2. A schematic for the method proposed by Franz et al. [21]. Descriptions are as in Figure 1. The PDE mass labelled α (orange) is the density of PDE mass that has flowed over the interface in the Brownian update-step. The peak in the PDE curve near the interface represents the addition of a δ-function corresponding to a Brownian particle that crosses the interface.

The Brownian time-step in the algorithm is required to be small, in order that the total probability of initialising a particle in the Brownian regime is less than one. However, the algorithm runs into difficulties if the time-step is chosen to be too small. Specifically, the amount of mass that flows over the interface between updates of the Brownian dynamics is too small in comparison to that which would be predicted theoretically using the exact diffusion kernel. This gives rise to inaccuracies in the algorithm, particularly if long simulation times are required. This sensitivity to the choice of Brownian time-step restricts the physical scenarios to which the algorithm can be applied.

In figure 3 we present three snapshots of the evolution of the first version of the algorithm (interface rather than overlap region) which illustrate this problem. By time $t = 2$, in Figure 3 (c), there is a clear disparity between the hybrid method and the mean field solution (black dotted line). Disparities of this nature are not acceptable when modelling real reaction-diffusion systems, irrespective of the computational savings the algorithm is able to produce.

3 The auxiliary region method

In this section we present our novel "auxiliary region method" (ARM) for coupling PDE and Brownian-based representations of reaction-diffusion. For simplicity we will present a version of the method with a single interface separating two regimes. However, the method can be easily generalised to multiple interfaces which separate alternating PDE and Brownian regions. Sequentially, we describe the composition of the domain and the models we employ in each region; the nature of the auxiliary regions; the implementation of movement of mass across the boundary; the implementation of reactions; and finally the



(a) (b) (c)

Figure 3. The evolution of 100 particles worth of mass initialised at a single PDE point at position $x = -0.95$, at times (a) 0.2, (b) 1 and (c) 2 under the first hybrid method of Franz et al. [21]. The green line is the PDE part of the hybrid method, the yellow bars represent the Brownian dynamics for the hybrid method (appropriately binned for visualisation purposes), the red line is the interface and the black dashed line is the solution of the mean-field diffusion equation (equation (16)). Results shown are averaged over 100 repeats.

specific details required for the simulation of the algorithm, including pseudocode for its implementation. All code, which has been written in MATLAB, can be found in the electronic supplementary material online.

3.1 The domain composition

Recall that, for our coupling method, space is partitioned into two regions within which we use different modelling paradigms (PDE and Brownian dynamics) to simulate the underlying reaction-diffusion system. Separating the two regions is a point interface, over which particles can jump according to a compartment-based method.

Consider a one-dimensional domain[3] $\Omega = (L_1, L_2) \subseteq \mathbb{R}$ for some $L_1 < 0 < L_2$. We split Ω into two regions, $\Omega_{\text{P}} = (L_1, 0)$ and $\Omega_{\text{B}} = (0, L_2)$ (separated by an interface I at position 0), within which the evolution of the system will be represented using a PDE description and Brownian dynamics, respectively.

3.2 The auxiliary regions

Particles can move between the two domains (Ω_{P} and Ω_{B}) via the auxiliary regions Ω_{PA} and Ω_{BA}; subsets of Ω_{P} and Ω_{B} respectively, each of width $h_a > 0$. Within these regions, mass/particles are simultaneously represented according to the default methodology for their domain (either PDE in Ω_{P} or Brownian dynamics in Ω_{B}), but also as well-mixed particles in their respective auxiliary regions Ω_{PA} and Ω_{BA}. These auxiliary regions act as a bridge between the fine- and coarse-scale descriptions. A schematic representation of the domain's composition is given in Figure 4.

We justify the use of the Brownian auxiliary region by following the methodology set out in Flegg et al. [19]. The entire Brownian domain can be simulated using a mesoscopic compartment-based regime, and equivalently using a microscopic simulation. In the absence of reactions, if the particles in the microscopic simulation are "binned" into the same compartments as the mesoscopic simulation, the expected numbers

[3]Note that the method can be extended to higher dimensions with (hyper-)planar interfaces in a straight-forward manner.

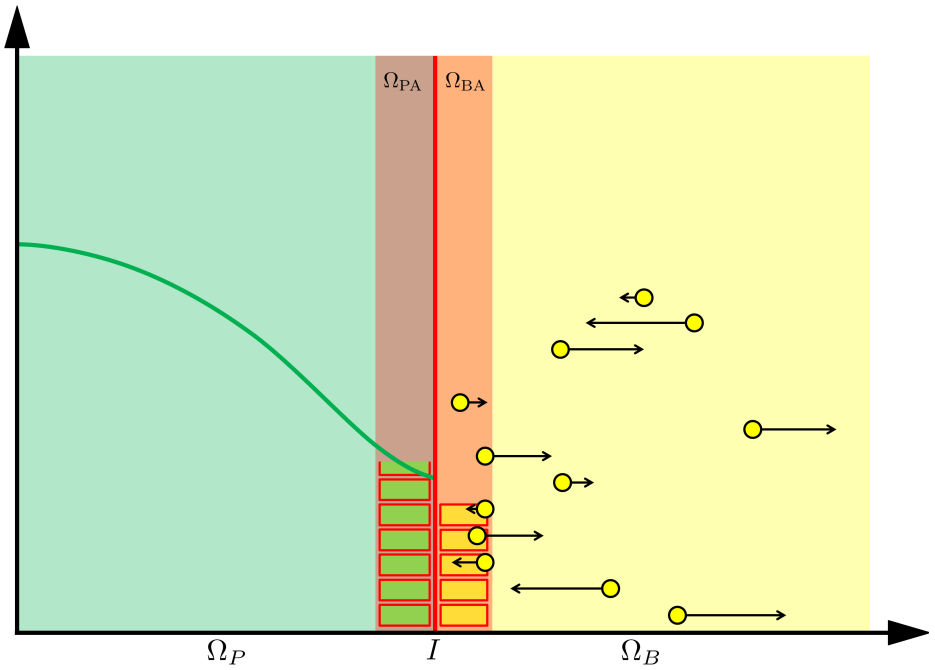


Figure 4. A schematic for the auxiliary region method (ARM). Descriptions as in Figure 1. The interface is the red line in the centre and the two auxiliary regions are shown with blocks to indicate the number of particles residing within them. In the PDE and Brownian auxiliary regions, each block represents a particle in the compartment-based representation and the number of blocks is determined by integrating the PDE over the auxiliary region Ω_{PA}, and counting the number of Brownian particles in Ω_{BA}, respectively.

in each compartment for each simulation would be the same. At this scale, the two methods are equivalent ways of simulating the same diffusive process [19].

To justify the use of the PDE auxiliary region, we appeal to the arguments of Yates and Flegg [49]. We note that the PDE density can be thought of as the probability of finding a particle at a particular position and time, scaled by the number of particles within the PDE domain. Provided that the auxiliary region is sufficiently narrow, the PDE density within the auxiliary region can be thought of as being approximately uniformly distributed across the region with the appropriate number of particles. This is precisely the interpretation of the contents of a compartment within the mesoscopic, compartment-based framework.

3.3 The PDE regime, Ω_{P}

Within Ω_{P}, we represent the mass of particles using:

$$\begin{aligned}
\text{PDE}: &\quad \frac{\partial \boldsymbol{c}}{\partial t}(x,t) = D\frac{\partial^2 \boldsymbol{c}}{\partial x^2}(x,t) + \boldsymbol{f}(\boldsymbol{c}(x,t)); \quad x \in \Omega_{\text{P}}; \quad t \in (0,T), \\
\text{BCs}: &\quad \frac{\partial \boldsymbol{c}}{\partial x}(x,t) = 0; \quad x \in \partial\Omega_{\text{P}}; \quad t \in (0,T), \\
\text{IC}: &\quad \boldsymbol{c}(x,0) = \boldsymbol{c}_0(x); \quad x \in \bar{\Omega}_{\text{P}}.
\end{aligned} \tag{1}$$

Here, $\boldsymbol{c}(x,t) = (c_1(x,t), ..., c_{\text{K}}(x,t))^T$, denotes the density of species $k = 1, \ldots, K$ at position x and time t, D is a diagonal matrix containing the Fickian diffusion constants for each species, and $\boldsymbol{f}$ is a function that encapsulates the effect of any reactions on each species. We also use the notation $\partial\Omega_{\text{P}}$ to represent the boundary of Ω_{P}, and $\boldsymbol{c}_0(x)$ is the initial condition. For all the simulations presented in this paper we employ the finite-difference θ-method (a general family of finite-difference methods)[4]. Although the Crank-Nicolson method ($\theta = 0.5$) is second-order accurate and unconditionally stable, we use $\theta = 0.51$ since the Crank-Nicolson method can give rise to spurious oscillations when implemented on step-function initial conditions of the sort we will consider [43].

3.4 The Brownian regime, Ω_{B}

Within Ω_{B}, all particles are tracked and their positions updated according to the following (computational) stochastic differential equation (SDE) which simulates Brownian motion:

$$y_i^k(t+\Delta t) = y_i^k(t) + \sqrt{2D_k\Delta t}\,\xi_i^k; \quad \xi_i^k \sim N(0,1); \quad \text{for} \quad i \in \{1, ..., N_{\text{HB}}^k(t)\} \quad \text{and} \quad k \in \{1, ..., K\}, \tag{2}$$

where $y_i^k(t)$ denotes the location of particle i of species k within Ω_{B}, Δt is the time-step for both the PDE and Brownian dynamics simulators[5] and $N_{\text{HB}}^k(t)$ is the number of particles of species k in Ω_{B} at time t. Once again, we set reflective boundary conditions at both ends of Ω_{B} to ensure that no particles can leave this domain via a Brownian diffusion event. The zero-flux boundary conditions at the interface for both PDE and Brownian regimes ensure that mass can only cross the interface according to the compartment-based method.

3.5 Movement across the interface

Since both domains, Ω_{P} and Ω_{B}, have zero-flux boundaries at the interface, particles can only cross over the interface via the auxiliary regions. In effect, these regions comprise a two-compartment reaction-diffusion master equation (RDME) model. Each particle in each auxiliary region jumps to its neighbouring region on the other side of the interface with a rate d_k (for species k), which is related to the macroscopic diffusion coefficient (for species k), D_k, via

$$d_k = \frac{D_k}{h_a^2}. \tag{3}$$

Here, h_a is the width of each auxiliary region, which is assumed to be the same for both the Brownian and PDE auxiliary regions. In order to implement jumps (or reactions, where necessary) according to the RDME, we require particle numbers.

Borrowing terminology from Yates and Flegg [49], the number of "pseudo-particles" of species k within the PDE auxiliary region, Ω_{PA}, at time t, denoted $N_{\text{PA}}^k(t)$, is calculated as

$$N_{\text{PA}}^k(t) = \int_{\Omega_{\text{PA}}} c_k(x,t)\; dx. \tag{4}$$

The number of particles of species k in the Brownian auxiliary region, Ω_{BA}, is given by

$$N_{\text{BA}}^k(t) = \left|\left\{j : y_j^k(t) \in \Omega_{\text{BA}}\right\}\right|. \tag{5}$$

[4]Note that this PDE can be simulated using any appropriate numerical solver, including the finite-element method or finite-volume method.

[5]Note that there is no requirement for the PDE and Brownian time steps to be the same. In many situation it may be useful to have a significantly finer Brownian time-step than PDE time step in order to accurately resolve the individual-based dynamics. We employ the same time-step in our simulations for simplicity.

These particle numbers allow us to define propensity functions corresponding to diffusive jumps between, or reactions within, the auxiliary regions. For diffusive jumps between the two auxiliary regions, the propensity functions for species k within the PDE and Brownian auxiliary regions are (respectively):

$$\alpha_{\mathrm{P}}^k(t) = d_k N_{\mathrm{PA}}^k(t) \quad \text{for} \quad \Omega_{\mathrm{PA}}, \tag{6}$$

$$\alpha_{\mathrm{B}}^k(t) = d_k N_{\mathrm{BA}}^k(t) \quad \text{for} \quad \Omega_{\mathrm{BA}}. \tag{7}$$

We note here that if $N_{\mathrm{PA}}^k(t) < 1$, we set $\alpha_{\mathrm{P}}^k(t) = 0$ to prevent the possibility of negative density. While it may be a problem if this scenario occurs persistently, practically speaking, we should choose the position of the interface such that density is always large enough that this does not happen. An adaptive interface will allow us to satisfy this criteria (see Section 4.4.3), and hence this problem would not occur when using such an interface.

When a particle jumps from Ω_{BA} to Ω_{PA}, a particle within the Brownian auxiliary region is chosen uniformly at random to be removed, and a particle's worth of mass is added to the PDE solution uniformly across Ω_{PA} for the species, k, which has changed:

$$c_k(x,t) = c_k(x,t) + \frac{1}{h_a}\mathbb{1}_{[x\in\Omega_{\mathrm{PA}}]}, \tag{8}$$

where $\mathbb{1}_{[x\in A]}$ is the indicator function for $x \in A$. Similarly, if a jump is enacted in the opposite direction, from Ω_{PA} to Ω_{BA}, we first remove a particle's worth of mass uniformly from Ω_{PA} for the appropriate species k:

$$c_k(x,t) = c_k(x,t) - \frac{1}{h_a}\mathbb{1}_{[x\in\Omega_{\mathrm{PA}}]}, \tag{9}$$

and a new particle is initialised within the Brownian auxiliary region, Ω_{BA}, with position chosen uniformly at random.

3.6 Reaction implementation

Throughout Ω_{P}, all reactions are implemented using the reaction operator $\boldsymbol{f}(\boldsymbol{c})$. The method we employ to implement reactions within Ω_{B} depends on the location of the reactant particles. Let $\mathcal{R}$ denote the set of reaction pathways (with $|\mathcal{R}| = R$). Define the subset of reactions $\mathcal{R}^*(t)$ at time t as follows:

$$\mathcal{R}^*(t) = \{\text{all reactions for which at least one set of reactant particles lies exclusively within } \Omega_{\mathrm{BA}}\}.$$

Reactions between molecules for which at least one of the reactive molecules lies within $\Omega_{\mathrm{B}}\backslash\Omega_{\mathrm{BA}}$ are implemented using an appropriate microscopic approach, such as the λ-ρ method [13, 33]. However, if at least one set of participating particles lie in Ω_{BA} (i.e. $r \in \mathcal{R}^*$), care needs to be taken over the interaction of such particles and the mass on the other side of the interface in Ω_{P}. As explained below we will implement the reactions $r \in \mathcal{R}^*$ for these reactant particles using the compartment-based method.

For illustrative purposes, consider a reversible second-order reaction involving species A, B and C:

$$A + B \underset{\kappa_2}{\overset{\kappa_1}{\rightleftharpoons}} C. \tag{10}$$

Under the $\lambda - \rho$ method [13] and its later modification [33], for the forward reaction, a particle of species A and a particle of species B are required to be within a distance ρ of one another in order to react. They then react with a rate λ, where λ is a function of both the reaction radius ρ and the reaction rate κ_1. Imagine that an A particle (without loss of generality) in Ω_{B} is close enough to the interface that the reaction radius ρ is larger than the distance between itself and the interface. For consistency with the Brownian representation, the A particle should be allowed to react with a B particle in the PDE region.

The implementation of such reactions would be extremely difficult. Instead, by ensuring bimolecular reactions within the auxiliary region are implemented according to the mesoscopic compartment-based method, we avoid such issues (provided that the width of the auxiliary region is chosen to be larger than the interaction radius ρ).

According to the backwards reaction, two particles are created after the reaction has occurred. These particles are placed a certain distance away from each other (called the dissociation radius) in order to achieve a specified probability of geminate recombination (a recombination of any pair of A and B particle that were initialised from the same C particle). If this radius intersects with the PDE regime, then there is the potential for individual particles to be initialised within Ω_{P}. By again employing the mesoscopic representations for reactions we resolve this issue. All product particles are assumed to be placed uniformly throughout the Brownian auxiliary region. Particles that are products of the backwards dissociation reaction in $\Omega_{\mathrm{B}}\backslash\Omega_{\mathrm{BA}}$ are extremely unlikely to be placed in Ω_{P} (again, providing that the auxiliary region is larger than the dissociation radius).

For these reasons, all of the reactions $r \in \mathcal{R}^*$ (for which at least one set of participating particles lie in Ω_{BA}) are implemented using the compartment-based method, in which reactions are incorporated as events in the associated Markov chain, according to the RDME. We can write the following propensity functions for reactions within Ω_{BA}:

$$\alpha_r(t) = g_r(\boldsymbol{N}_{\mathrm{BA}}(t))\kappa_r h_a^{1-\nu}, \tag{11}$$

for any reaction channel $r \in \mathcal{R}^*(t)$ of order ν and corresponding reaction rate κ_r, where $\boldsymbol{N}_{\mathrm{BA}}(t) = (N^1_{\mathrm{BA}}(t), ..., N^K_{\mathrm{BA}}(t))^T$ and g_r is the appropriate number of possible combinations of the reactants for reaction r from the particles that lie within Ω_{BA}. Recall, however, that in $\Omega_{\mathrm{B}}\backslash\Omega_{\mathrm{BA}}$, any such reactions are implemented according to the chosen microscopic reaction method [13, 33, 9].

3.7 Simulation specifics

The Gillespie SSA [24] is used to simulate the above-described reactions in Ω_{BA}, as well as the diffusive fluxes over the interface. The SSA requires the computation of an exponential random variable which gives the time, τ, until the next event, and can be found by transforming a uniform random variable $u \sim \mathrm{Unif}(0, 1)$ via the following equation

$$\tau = \frac{1}{\alpha^0(t)} \ln\left(\frac{1}{u}\right). \tag{12}$$

Here, $\alpha^0(t)$ is the sum of all of the propensity functions:

$$\alpha^0(t) = \alpha^0_{\mathrm{P}}(t) + \alpha^0_{\mathrm{B}}(t) + \sum_{r\in\mathcal{R}^*(t)} \alpha_r(t), \tag{13}$$

where

$$\alpha^0_{\mathrm{P}}(t) = \sum_{k=1}^{K} \alpha^k_{\mathrm{P}}(t), \tag{14}$$

and

$$\alpha^0_{\mathrm{B}}(t) = \sum_{k=1}^{K} \alpha^k_{\mathrm{B}}(t). \tag{15}$$

The PDE solutions and Brownian dynamics are implemented using the same discrete time-step, Δt, and the diffusive jumps across the interface (and any required reactions, $r \in \mathcal{R}^*$) are implemented in an event-driven manner, according to the Gillespie SSA. Event-driven time-steps are implemented until the putative time for the next event passes the next Brownian/PDE update time, at which point the PDE and Brownian dynamics are updated. Pseudocode for the ARM is given in Algorithm 1.

Algorithm 1: Auxiliary region method (ARM)

(1a) Initialise time $t = 0$, set final time T, PDE/Brownian update time-step, Δt, the PDE discretisation grid size, Δx, and the auxiliary region spatial step, h_a. Initialise particles in both Ω_{P} and Ω_{B} as required. Calculate the time until the next PDE and Brownian update step $t_\Delta = \Delta t$.

(1b) Calculate the number of particles N^k_{PA} and N^k_{BA} in the auxiliary regions, for each species $k \in \{1, 2, ..., K\}$, using formulae (4) and (5). Consequently, calculate the corresponding propensity functions, $\alpha^k_{\mathrm{P}}(t)$ and $\alpha^k_{\mathrm{B}}(t)$ as per equations (6) and (7), and their sums according to equations (14) and (15). Calculate $\alpha_r(t)$, for $r \in \mathcal{R}^*$, using equation (11) and finally compute $\alpha^0(t)$ according to equation (13).

(1c) Calculate the time, τ, until the next auxiliary region event according to equation (12). Update the auxiliary region time $t_a = t + \tau$.

(1d) If $t_a < t_\Delta$

(i) Draw three random numbers $u_1, u_2, u_3 \sim \mathrm{Unif}(0, 1)$.

(ii) If $u_1\alpha^0(t) < \alpha^0_{\mathrm{P}}(t)$ (corresponding to a jump from Ω_{PA} to Ω_{BA}):

- Use u_2 to determine the species, k, which the jump affects, with each species selected with probability proportional to its propensity function.
- Remove a particle from the PDE auxiliary region for species k via equation (9).
- Initialise a new particle of species k within Ω_{BA} at position $y^* = u_3 h_a + I$.

Else if $u_1\alpha^0(t) < \alpha^0_{\mathrm{P}}(t) + \alpha^0_{\mathrm{B}}(t)$ (corresponding to a jump from Ω_{BA} to Ω_{PA}):

- Use u_2 to determine the species, k, which the jump affects, with each species selected with probability proportional to its propensity function.
- Choose a particle of species k uniformly at random from within the Brownian auxiliary region and remove it from the system. We do this by selecting an index q such that $q = \lceil u_3 N^k_{\mathrm{BA}} \rceil$, where $\lceil x \rceil$ denotes the smallest integer larger than x.
- Add a new particle into the PDE auxiliary region for species k via equation (8).

Else (corresponding to a reaction in Ω_{BA})

- Use u_2 to choose the reaction $r \in \mathcal{R}^*(t)$ to be implemented with probability proportional to its propensity function.
- Update particle numbers (and initialise positions, if appropriate) in the Brownian representation accordingly.

(iii) Set $t = t_a$.

Else

(i) Update the PDE system (1) using an appropriate numerical method (see Section 3.1).

(ii) Update the positions of the Brownian particles according to equation (2).

(iii) Implement any reactions using an appropriate method (see Section 3.1). Note that production reactions should be implemented after any degradation reactions in order to prevent particles being created and destroyed in the same time-step.

(iv) Set $t = t_\Delta$, update $t_\Delta = t + \Delta t$.

(1e) If $t < T$, return to **(1b)**, otherwise stop.

4 Results

Within this section we present four test problems which are used to demonstrate that the ARM correctly simulates reaction-diffusion systems. Two of these problems are models of pure diffusion with different initial conditions and will demonstrate that the fluxes over the interface are consistent with the expected behaviour of the fully Brownian simulations. The third problem is the formation of a morphogen gradient, which demonstrates the successful implementation of reactions in the ARM. Despite the fact that our method is valid for higher-order reactions, the first three test problems consider reactions up to first order. For such systems, no moment closure assumptions are required in deriving the mean-field reaction-diffusion PDE and hence its behaviour agrees with the mean behaviour of the individual-based models. This allows us to efficiently verify accuracy by comparing the mean behaviour of our hybrid method to the known mean-field behaviour. Finally, in test problem four, we implement a second-order reaction system in higher dimensions, indicating the applicability of the method to more complicated examples.

For each of the first three test problems, we use $\Omega_{\text{P}} = (-1, 0)$ and $\Omega_{\text{B}} = (0, 1)$, meaning that the interface is the single point at 0. We take the value of the fixed PDE and Brownian time update steps to be $\Delta t = 0.02$, the auxiliary regions have width $h = 0.05$ and the diffusion constant is $D = 0.0025$ (unless specified otherwise). We will quantify the qualitative comparisons, presented in this section through density comparison snapshots, in Section 5. All simulations will comprise only a single species, so henceforth, all sub- or super-scripts, k, pertaining to species will be removed.

4.1 Test problem 1: maintaining equilibrium

For the first test problem, we simulate pure diffusion in the form of a simple Brownian motion with reflecting boundary conditions, which has Fokker-Planck equation given by the diffusion PDE and corresponding boundary conditions:

$$\text{PDE}: \quad \frac{\partial p}{\partial t} = D\frac{\partial^2 p}{\partial x^2}; \quad x \in (-1, 1); \quad t \in (0, T), \tag{16}$$

$$\text{BCs}: \quad \frac{\partial p}{\partial x}(x, t) = 0; \quad x = -1, 1; \quad t \in (0, T), \tag{17}$$

$$\text{IC}: \quad p(x, 0) = p_0(x); \quad x \in [-1, 1], \tag{18}$$

where $p_0(x)$ denotes the initial condition. Note that $p(x, t)$ here represents the mean-field solution across the whole domain, whereas $c(x, t)$ represents the PDE solution in Ω_{P} in the hybrid method. We initialise particles uniformly across the computational domain, so that $p_0(x) \equiv N/2$, where N is the (constant) number of particles in the system. Figure 5 shows that the ARM passes the most basic test by maintaining the steady state without causing an accumulation of mass on either side of the interface. For this test problem, we also include a plot which displays the variance in the density of particles (Figure 6). In order to calculate this variance, we have binned the spatial domain onto a mesh of size h_a (the same as the auxiliary region width) and calculated the variance of the density in each bin over a number of identically initialised (up to random allocation of particles in Ω_{B}) repeats. This demonstrates a problem that occurs with all hybrid methods which contain an interface coupling a stochastic to a deterministic region. The variance is damped close to the interface in the stochastic part of the domain, due to the deterministic nature of the solver on the opposite side. Specifically, the PDE effectively has a stochastic boundary condition at the interface, caused by the diffusive jumps between the auxiliary regions. This causes a higher level of variance than would be expected if it was a purely deterministic regime. However, when a particle jumps from the PDE to the Brownian dynamics auxiliary region, since the PDE region is mostly deterministic, it contributes less variance than would be expected than if the stochastic method was employed across the entire domain. There are methods that can be used in order to fix this problem, such as the use of an overlap region (e.g. [28]) and replacing the PDE with an appropriate SPDE (e.g. [1]). This is explored in more detail in the discussion (Section 6).

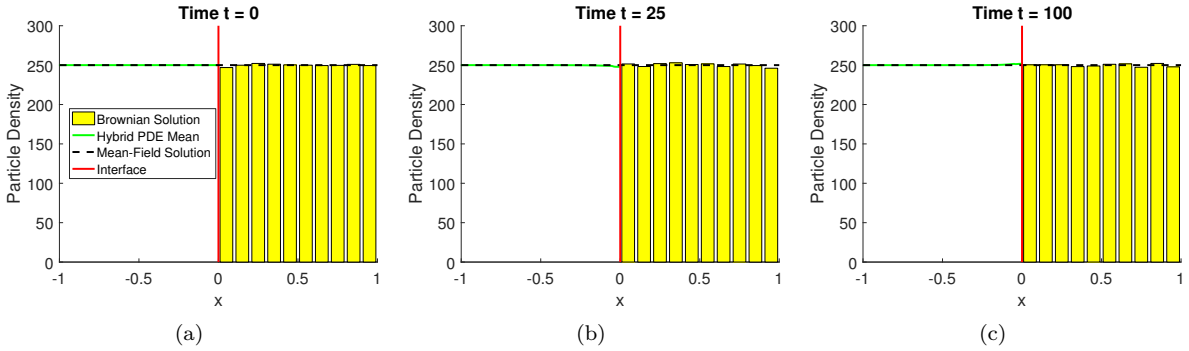


Figure 5. The evolution of test problem 1 at times (a) 0, (b) 25 and (c) 100. The green line is the PDE part of the hybrid method, the yellow bars represent the Brownian dynamics for the hybrid method (appropriately binned for visualisation purposes), the red line is the interface and the black dashed line is the solution of the mean-field model (16) under the given initial condition. Results shown are for $N = 500$ particles and are averaged over 1000 repeats. We solve the PDE with the θ-method, with a value of $\theta = 0.51$. All other parameters are given within the text.

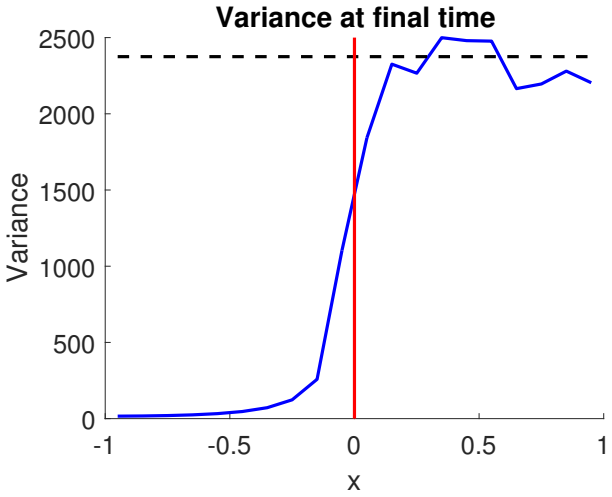


Figure 6. The plot of the variance in the density of particles at time $t = 100$ for the parameter values used to produce Figure 5. The spatial domain is partitioned into a series of bins of width h_a, and the particle density variance is calculated in each bin over $S = 1000$ repeats. The blue line is the variance from the hybrid method, the black dashed line is the expected variance from the fully Brownian model, and the red line is the position of the interface. The variance can be seen to be damped in the stochastic domain close to the interface, as discussed in the text.

4.2 Test problem 2: flux over the interface

The second test problem is a stress test for the interfacial flux. For the PDE part of the hybrid method we solve the same diffusion equation (16)-(18) as in Section 4.1. However this time we initialise by placing

all particles uniformly within the PDE domain, Ω_{P}, which results in

$$p_0(x) = \begin{cases} N & x \in \Omega_{\mathrm{P}} \\ 0 & x \in \Omega_{\mathrm{B}} \end{cases}$$

The results from this simulation are displayed in Figure 7. As with the uniform initial condition in test

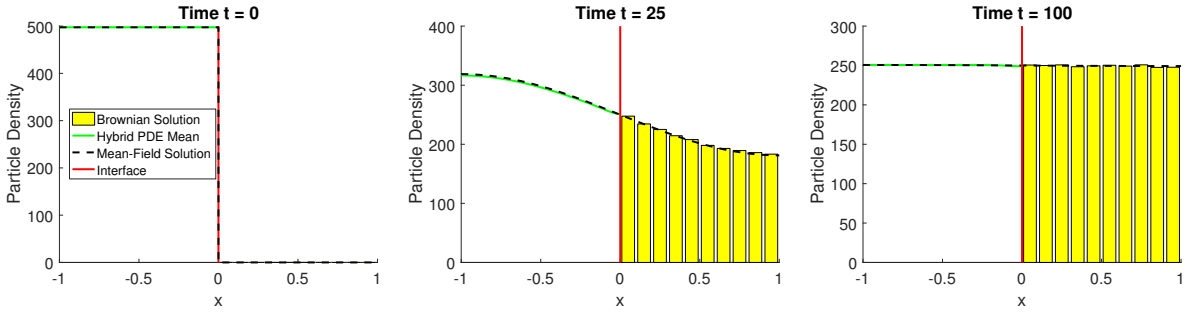


Figure 7. The evolution of test problem 2 at times (a) 0, (b) 25 and (c) 100. We use $D = 0.025$ and $\Delta t = 0.005$. All figure descriptions and other parameter values are as in figure 5.

problem 1, we see from Figure 7 that the hybrid method agrees with the solution of the mean-field model, indicating that the method simulates flux over the interface accurately. We have also tested our hybrid method with all the mass initialised uniformly across Ω_{B} and found a similarly good agreement between the hybrid method and the mean-field solution (figures not shown).

4.3 Test problem 3: morphogen gradient

For the third test problem, we investigate the formation of a morphogen gradient from a uniform initial condition. The gradient is formed by allowing particles to diffuse throughout the domain as well as to degrade at a rate μ. We also have particles entering at the left-hand boundary, $x = -1$, at rate $D\lambda$, and a zero-flux condition at $x = 1$. Thus, the PDE half of the hybrid domain is governed by the mean-field model representing the expected behaviour of the fully Brownian dynamics:

$$\begin{aligned} \text{PDE}: \quad & \frac{\partial c}{\partial t} = D\frac{\partial^2 c}{\partial x^2} - \mu c; \quad x \in (-1, 0); \quad t \in (0, T), \\ \text{BCs}: \quad & \frac{\partial c}{\partial x}(-1, t) = -\lambda; \quad \frac{\partial c}{\partial x}(0, t) = 0; \quad t \in (0, T), \\ \text{IC}: \quad & c(x, 0) = c_0(x); \quad x \in [-1, 0]. \end{aligned} \tag{19}$$

For the corresponding microscopic dynamics we implement Brownian motion for the diffusion of particles and a time-based method in order to enact the degradation reactions. We note that production of particles is not implemented within the microscopic domain since it occurs at $x = -1$. $N = 500$ particles are initialised uniformly across the whole domain $[-1, 1]$.

As demonstrated in Figure 8 the solution of the hybrid method matches that of the corresponding mean-field model, as with the previous two test problems.

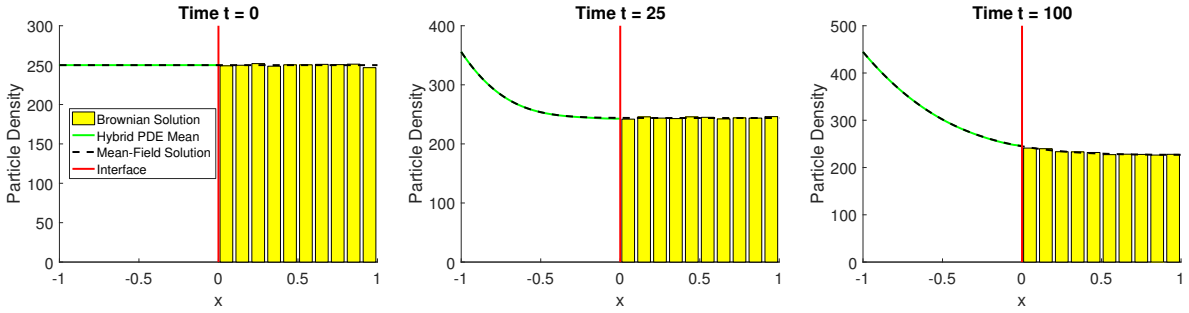


Figure 8. The evolution of test problem 3 at times (a) 0, (b) 25 and (c) 100. The value for the production rate is $\lambda = 400$ and the degradation rate is $\mu = 0.001$. All other figure descriptions and parameter values are the same as Figure 5.

4.4 Test problem 4: higher-order systems

For our final test problem, we look at the reaction system:

$$2A \xrightarrow{\kappa_1} \emptyset, \quad \emptyset \xrightarrow{\kappa_2} A, \tag{20}$$

which takes place in a three-dimensional cuboid $\Omega \subseteq \mathbb{R}^3$ of volume V, where $\Omega = (x_0, x_1) \times (y_0, y_1) \times (z_0, z_1)$. We divide this domain by firstly defining the position of the adaptive planar interface $I(t) \in (x_0, x_1)$ which is to be employed in this test problem (see Section 4.4.3). In an analogous way to in the one-dimensional case, we then define the now time-dependent PDE and individual-based subdomains, $\Omega_{\mathrm{P}}(t)$ and $\Omega_{\mathrm{B}}(t)$, with volumes $V_{\mathrm{P}}(t)$ and $V_{\mathrm{B}}(t)$ respectively. These subdomains and volumes depend on t due to the adaptive interface position. The interface will move according to the local density profile within the PDE and Brownian dynamics auxiliary regions $\Omega_{\mathrm{PA}}(t)$ and $\Omega_{\mathrm{BA}}(t)$, which are explicitly defined to be:

$$\Omega_{\mathrm{PA}}(t) = (I(t) - h_a, I(t)) \times (y_0, y_1) \times (z_0, z_1),$$
$$\Omega_{\mathrm{BA}}(t) = (I(t), I(t) + h_a) \times (y_0, y_1) \times (z_0, z_1).$$

Before specifying how the interface will move, we will firstly find a PDE in one dimension that we will use to simulate the deterministic part of our system. We do this by considering the reaction system (20) and forming an ODE in three dimensions. We then include isotropic diffusion to obtain a three-dimensional PDE, and finally impose a constraint on the initial condition to simplify this to a one-dimensional PDE. We then briefly describe the process we use to evolve the individual-level behaviour, before introducing an adaptive interface. We will finish this subsection with the results of some simulations of this system. Note that from now on, we will drop the dependence on t for any of the subdomains, their volumes and the interface position for brevity, unless they are explicitly needed.

4.4.1 PDE model

We will use the chemical master equation (CME) for the reaction system in order to derive a PDE that approximates the system (20) in Ω_{P}. Let $p_n(t) = \mathbb{P}(A(t) = n)$, where $A(t)$ is the number of particles at time t. Then the CME for the evolution of this probability is given by:

$$\frac{\mathrm{d}p_n}{\mathrm{d}t} = \frac{\kappa_1}{V_{\mathrm{P}}} \left[(n+2)(n+1)p_{n+2} - n(n-1)p_n\right] + \kappa_2 V_{\mathrm{P}} \left[p_{n-1} - p_n\right].$$

If we now define the k^{th} central moment $\langle A^k \rangle := \sum_{n=0}^{\infty} n^k p_n$, we can multiply the CME by n and sum over all $n \in \mathbb{N}_0$ to yield the mean equation:

$$\frac{\mathrm{d}\langle A \rangle}{\mathrm{d}t} = \frac{2\kappa_1}{V_{\text{P}}} \left[\langle A \rangle - \langle A^2 \rangle \right] + \kappa_2 V_{\text{P}}. \tag{21}$$

The ODE (21) is currently exact, but depends on the second moment of A. Furthermore, the ODE for every moment of A depends on higher moments still — the system is not closed. In order to close the system, we follow Erban and Chapman [13] and apply Poisson moment closure, which implies:

$$\mathbb{V}\mathrm{ar}(A) = \mathbb{E}[A] \Longrightarrow \langle A^2 \rangle = \langle A \rangle + \langle A \rangle^2. \tag{22}$$

Applying the moment closure (22) to the ODE (21), and setting $c = \langle A \rangle / V_{\text{P}}$ gives us the closed ODE

$$\frac{\mathrm{d}c}{\mathrm{d}t} = \kappa_2 - \kappa_1 c^2.$$

Finally, including isotropic diffusion through the usual Laplace operator yields the three-dimensional PDE:

$$\frac{\partial c}{\partial t} = D\nabla^2 c - \kappa_1 c^2 + \kappa_2; \quad (x, y, z) \in \Omega; \quad t \in (0, T). \tag{23}$$

We will enforce an initial condition which is translationally invariant in both the y and z co-ordinates, which means that the dynamics will remain translationally invariant for all time. As such, c is simply a function of x and t, and the dynamics can be represented by a one-dimensional equivalent of this PDE by implementing zero-flux conditions on all boundaries and using the transformation:

$$\bar{C}(x,t) = \int_{z_0}^{z_1} \int_{y_0}^{y_1} c(x,t)\ dy\ dz = L_y L_z c(x,t),$$

where $L_y = y_1 - y_0$ and similarly for $L_z = z_1 - z_0$. This gives:

$$\begin{aligned}
\text{PDE}: &\quad \frac{\partial \bar{C}}{\partial t} = D\frac{\partial^2 \bar{C}}{\partial x^2} - \frac{\kappa_1}{L_y L_z}\bar{C}^2 + \kappa_2 L_y L_z; \quad x \in (x_0, I); \quad t \in (0, T), \\
\text{BCs}: &\quad \frac{\partial \bar{C}}{\partial x}(x,t) = 0; \quad x = x_0, I; \quad t \in (0, T), \\
\text{IC}: &\quad \bar{C}(x,0) = \bar{C}_0(x); \quad x \in [x_0, I].
\end{aligned} \tag{24}$$

4.4.2 Individual-based formulation

We now turn our attention to the individual-based system. In order to simulate the three-dimensional individual-based model, we will follow the λ-ρ method [13]. In the context of this system, whenever two particles are within the reaction radius ρ, they react with a probability P_λ, which is a function of the kinetic rate κ_1, the time-step Δt, and the diffusion coefficient D. For more information on how P_λ is chosen, we refer the reader to Erban and Chapman [13]. The zeroth-order reaction is completed by initialising a particle uniformly throughout the individual-based domain Ω_{B} with probability $\kappa_2 \Delta t V_{\text{B}}$, which we ensure is below 1 by choosing Δt to be sufficiently small.

4.4.3 Adaptive interface

Test problems 1–3 have been simulated using a static interface. However, this requires *a priori* knowledge of where the interface should be for all time. When the finer scale modelling regime is required in order to resolve a fixed area of space in more detail (for example, the region around ion channels [8]), the interface

position may be known. However, if the purpose of the interface is to split regions of space in which there are high and low particle numbers in situations in which particle numbers change dynamically, a different approach is required. In this case, the interface (or interfaces) need to move with the density of particles to maintain the computational savings they are designed to provide. We now describe a method, adapted from Robinson et al. [39] which allows the interface to move adaptively.

The interface at time t, which we shall denote by $I(t)$, moves according to local particle numbers in the auxiliary regions around it. We set two thresholds $\beta_u > \beta_l$, and move the interface towards the PDE subdomain if $N_{\text{PA}}(t) < \beta_l$ and towards the individual-based subdomain if $N_{\text{BA}}(t) > \beta_u$ (borrowing the notation from Section 3). The two threshold values are designed to prevent the interface from rapidly oscillating between two values, which is a possibility when $\beta_u = \beta_l$ due to the stochastic nature of the system. We enforce that when the interface moves, it moves a distance h_a, the width of the auxiliary region, in the chosen direction.

If the interface moves towards the PDE subdomain (i.e. $N_{\text{PA}}(t) < \beta_l$), we convert the PDE auxiliary region into particles, initialising each one uniformly across the new Brownian auxiliary region Ω_{BA}. As $N_{\text{PA}}(t)$ is not necessarily an integer, we treat the fractional part ($N_{\text{PA}}(t)$ mod 1) to be the probability of initialising one extra particle within the newly formed individual-based region. We then scale the rest of the PDE subdomain to ensure that we conserve mass. During an interface movement towards the individual-based subdomain (i.e. $N_{\text{BA}}(t) > \beta_u$), the Brownian auxiliary region is converted to PDE mass by initialising a density of $N_{\text{BA}}(t)/h_a$ uniformly across the new PDE auxiliary region, Ω_{PA}, created by moving the interface. For a more detailed description of a similar method, we direct the interested reader to Robinson et al. [39].

4.4.4 Results

We consider N particles initialised across Ω with a constant negative gradient so that the density of particles at position x_1 is equal to zero. This ensures that the interface will move as the dynamics progress. The results can be seen in Figure 9, in which the hybrid method has been averaged over $S = 1000$ repeats. The hybrid density in the case of the moving interface is represented as yellow bars throughout the domain. This is because the interface position changes with each repeat, and so very few regions of space are solely represented by one or the other modelling paradigm over all repeats.

Space		**Experimental**		**Model**	
x_0	0	D	0.2	Δt	0.01
x_1	10	κ_1	0.01	h_p	0.1
y_0	0	κ_2	0.5	h_a	0.5
y_1	2	ρ	0.1	I_0	0.5
z_0	0			N	200
z_1	2			T	5
V	40			β_u	9.5
				β_l	4

Table 2. Table of parameter values for test problem 4.

We can see good agreement between the hybrid method and the fully individual-based method throughout the domain, with the only discrepancy close to the left hand boundary at 0 caused by the difference between the PDE and individual-based methods due to moment closure. We compare our hybrid method to the fully individual-based method here, in contrast to the PDE solution used in test problems 1-3, due to the inaccuracy introduced in the PDE by the moment closure required for the second order reaction.

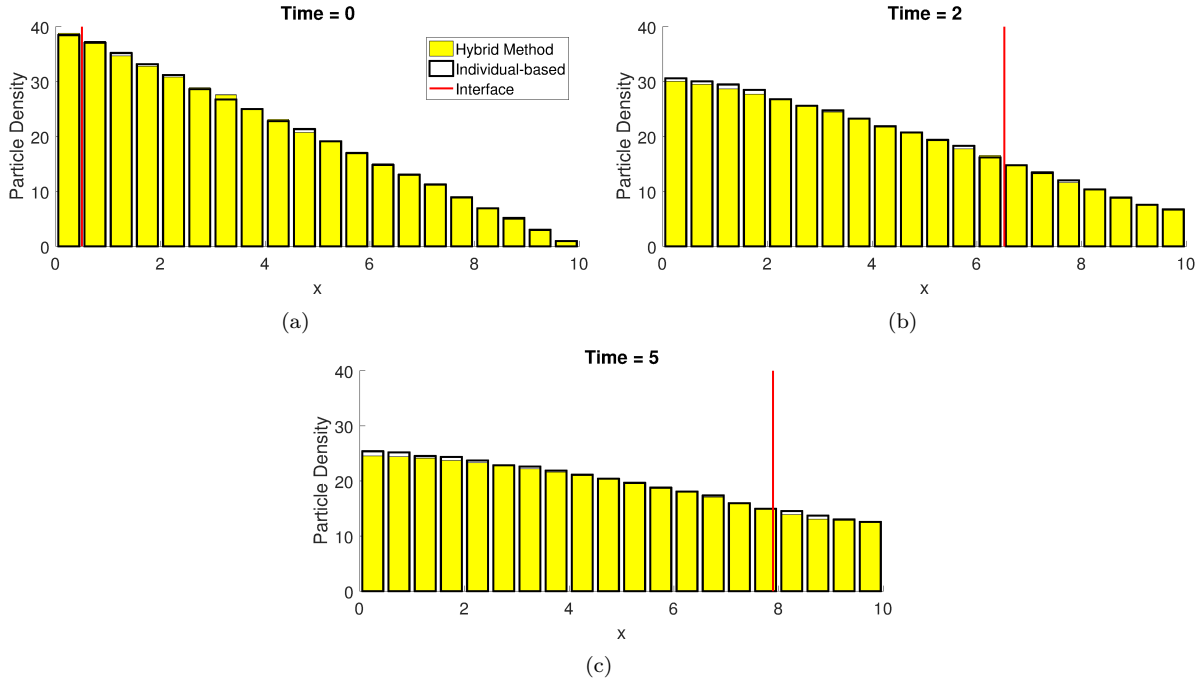


Figure 9. The evolution of test problem 4 at times (a) 0, (b) 2 and (c) 5. The yellow bars represent the hybrid solution, binned onto a mesh with width h_a, and the black outline bars are the fully individual-based solution, which has been binned onto the same mesh as the hybrid solution. The vertical red line is the average position of the interface over the $S = 1000$ repeats. All parameters are as in Table 2.

5 Error analysis

We have seen in Section 4 that the solutions provided by the hybrid method visually match the mean-field solution. Within this section we quantify the difference using the solutions of these test problems. We compare the mass in the PDE and Brownian regions of the domain between the two methods. Separately we compare the density profile across the whole domain using the histogram distance error (HDE). We then proceed to investigate the dependence of the accuracy of the hybrid method on the two free simulation parameters (Δt and h_a).

5.1 Quantitative comparisons

In order to evaluate the accuracy of the ARM for test problems 1,2 and 3, we compare its mean behaviour (averaged over $S = 1000$ repeat simulations) to the mean-field model for which we compute the analytical solution across the entire domain Ω, for each of our test problems. Figure 10 contains nine plots which demonstrate the error for the first three test problems above; (a)-(c) are for test problem 1, (d)-(f) are for test problem 2 and (g)-(i) are for test problem 3. The first and second columns show particle number comparisons between the hybrid and analytical solutions. Specifically, in the first column we compare

$$N_{\text{MP}}(t) = \int_{-1}^{0} p(x,t)\mathrm{d}x, \tag{25}$$

the expected number of particles in Ω_{P} in the mean-field model to

$$N_{\text{HP}}(t) = \frac{1}{S}\sum_{s=1}^{S}\int_{-1}^{0} c^s(x,t)\mathrm{d}x, \tag{26}$$

the expected number of particles in Ω_{P} in the hybrid method. Here, as before, $p(x,t)$ represents the mean-field PDE solution at position x at time t and $c^s(x,t)$ represents the PDE part of the solution in the hybrid method for repeat s of S. Explicitly, we plot $(N_{\text{HP}} - N_{\text{MP}})/N_{\text{MP}}$, which shows no bias around zero for any of the three test problems. For completeness, in the second column we also compare

$$N_{\text{MB}}(t) = \int_{0}^{1} p(x,t)\ \mathrm{d}x, \tag{27}$$

the expected number of particles in Ω_{B} in the mean-field model to

$$N_{\text{HB}}(t) = \frac{1}{S}\sum_{s=1}^{S} N^s_{\text{HB}}(t), \tag{28}$$

the expected number of particles in Ω_{B} in the hybrid methods. Here, $N^s_{\text{HB}}(t)$ is the number of particles in the Brownian region of the hybrid method at time t for repeat s of S. Explicitly, we plot $(N_{\text{HB}} - N_{\text{MB}})/N_{\text{MB}}$, which again shows no bias around zero for any of the three test problems.

The last column of Figure 10 contains the histogram distance error (HDE), which is defined by

$$\text{HDE}(t) = \frac{1}{2}\sum_{\ell=1}^{L}\left|c^H_\ell(t) - c^P_\ell(t)\right|, \tag{29}$$

where ℓ indexes a common mesh on which the solutions are compared. $c^H_\ell(t)$ is the normalised solution of the hybrid method at mesh point ℓ and time t, and $c^P_\ell(t)$ is the normalised solution of the mean-field model at the same common mesh point and time, where

$$\sum_{\ell=1}^{L} c^H_\ell(t) = \sum_{\ell=1}^{L} c^P_\ell(t) = 1 \quad \forall t \geq 0.$$

This ensures a value of the HDE between 0 and 1. Here, 0 means that the two solutions are exactly the same, and 1 corresponds to the two solutions having non-overlapping supports. All figures were produced using the same number of repeats ($S = 1000$).

In all cases, the relative errors between the mean-field and hybrid methods, in Figure 10, are low with no discernible bias about zero. Similarly, all HDE plots in Figure 10 are low for the majority of the simulations. This demonstrates numerically that the hybrid scheme presented in this paper is correctly reproducing the behaviour of the Brownian model in the mean-field. These error plots confirm the visual concurrence shown in Figures 5–8.

For the fourth test problem, we use a different error measurement due to the disparity between the mean-field PDE and individual-based systems. Consequently, we choose to compare the number of particles in the final compartment for both the hybrid method and the individual-based method. We motivate this in two ways. Firstly, using this measure of error, we are able to minimise the influence of the extra error caused by the difference between the mean-field PDE and the individual-based method. Secondly, several biological systems require detailed knowledge of the particle concentrations at the end of the domain. Apical growth of filamentous cells such as fungi [27] is such an example. If we define $N_{\text{H}}(t)$ to be the average number of particles in the region $(x_1 - h_a, x_1) \times (y_0, y_1) \times (z_0, z_1)$ (where we recall that

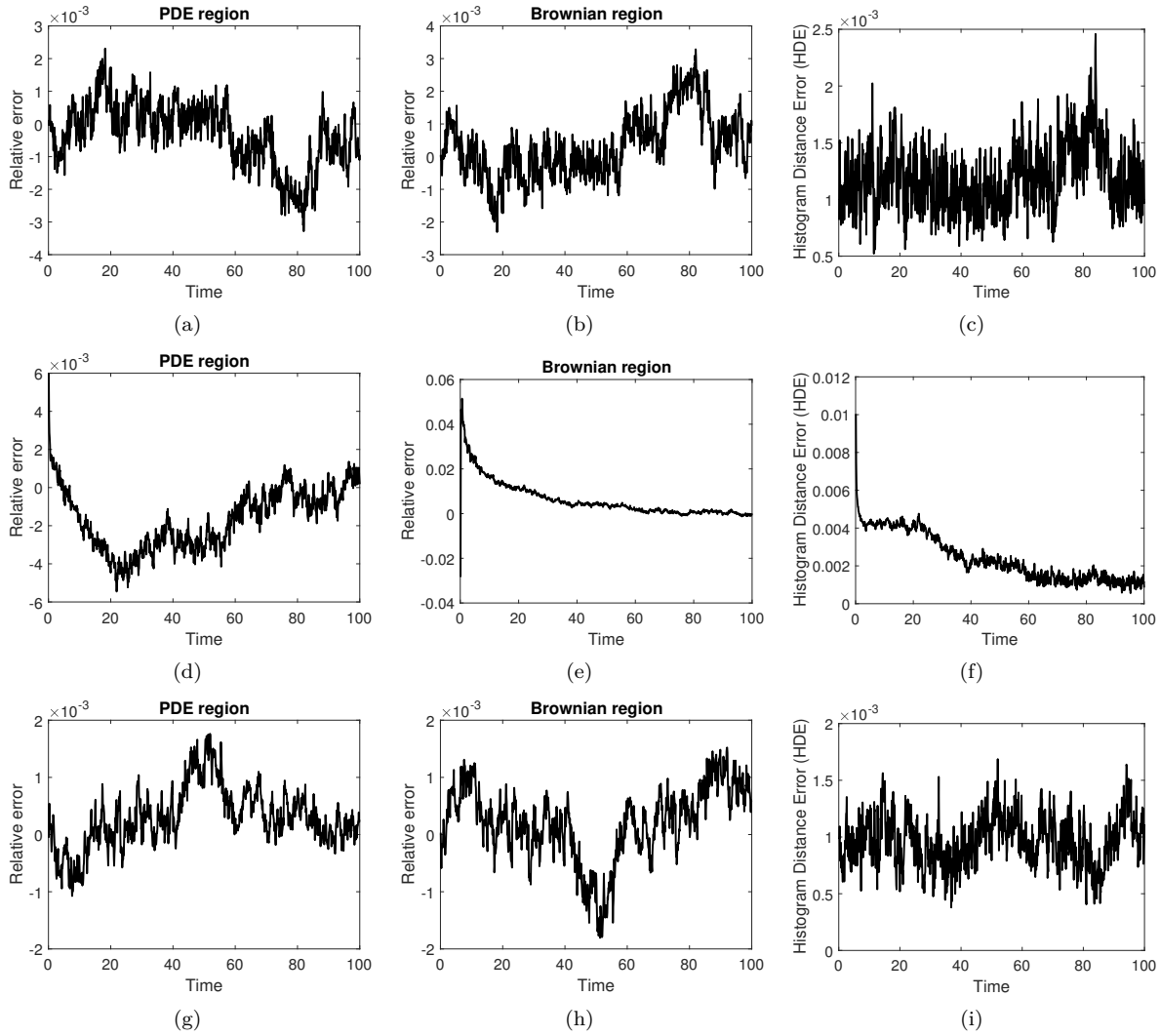


Figure 10. Error plots for test problems 1 (a)-(c), 2 (d)-(f) and 3 (g)-(i). The first column contains the relative errors $(N_{\text{HP}} - N_{\text{MP}})/N_{\text{MP}}$, while the second column contains the relative errors $(N_{\text{HB}} - N_{\text{MB}})/N_{\text{MB}}$, and the third column contains the HDE comparison using equation (29).

$\Omega = (x_0, x_1) \times (y_0, y_1) \times (z_0, z_1)$), when simulating the hybrid method at time t, and the quantity $N_{\text{M}}(t)$ to be the same for the fully microscopic simulation, we can obtain a measurement of error given by

$$E_{\text{Rel}}(t) = \frac{N_{\text{M}}(t) - N_{\text{H}}(t)}{N_{\text{M}}(t)}. \tag{30}$$

The relative error shows no long-term bias in either direction, and oscillates around zero, indicating a close agreement between our hybrid method and the ground truth individual-based method. The hybrid method completed 1000 repeats in 485.5 seconds, while the fully individual-based method took 1047.4 seconds.

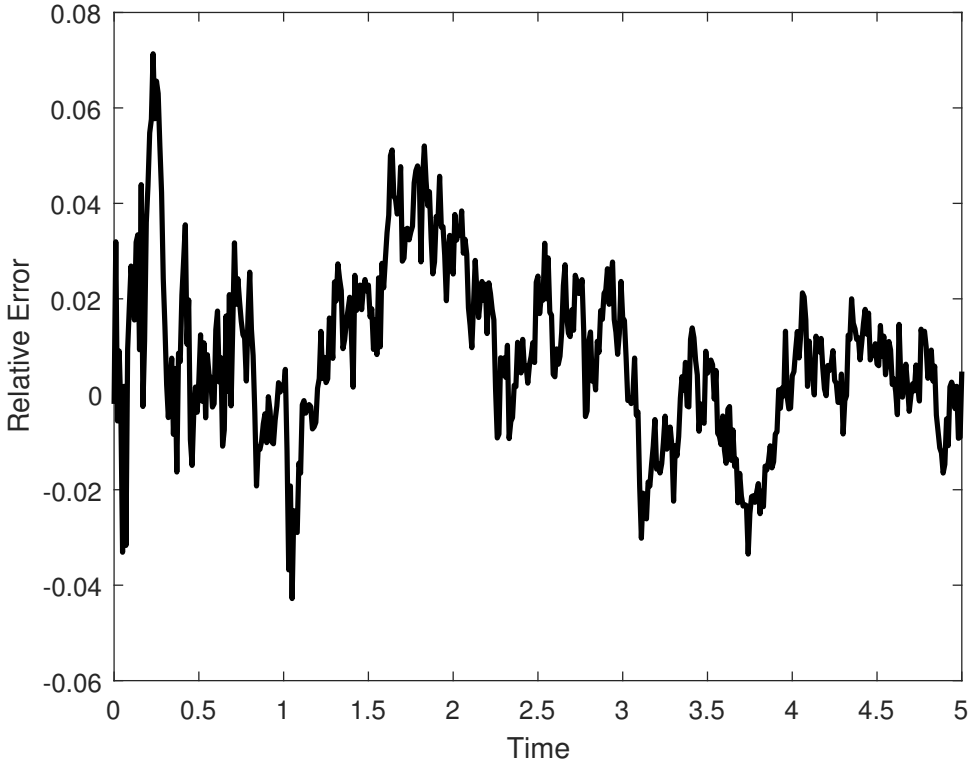


Figure 11. The error measurement given in equation (30) for the system simulated in Figure 9.

5.2 Parameter choice

Within the ARM, there are two free parameters that need to be chosen – the width of the auxiliary regions h_a and the time-step for the PDE and Brownian updates Δt. These need to be chosen so that the quantity $D\Delta t/h_a^2$ remains small enough that the particle numbers in the auxiliary regions do not become overly equilibrated between PDE/Brownian update steps. That is to say, if there is a gradient across the interface, Δt should be small enough that the closed system of the two auxiliary regions should not reach steady state between PDE/Brownian update steps.

In order to demonstrate why $D\Delta t/h_a^2$ must be small, we consider the evolution of particle numbers in the two auxiliary regions between PDE/Brownian update steps. We form an ODE for particle numbers in one of these boxes (using the fact that particle numbers are conserved between PDE/Brownian updates).

Let ν_0 be the (constant) number of particles in the two auxiliary regions combined, $M_{\text{P}}(t)$, $M_{\text{B}}(t)$ be the mean number of particles in the PDE and Brownian auxiliary regions respectively at time t, and μ_{P}, μ_{B} be the number in the PDE and Brownian auxiliary regions respectively at time 0, which will represent the beginning of a time-step. Then, the equation for the mean number of particles in the PDE auxiliary region can be calculated from a simple probability master equation as

$$\frac{\mathrm{d}M_{\text{P}}}{\mathrm{d}t} = dM_{\text{B}} - dM_{\text{P}} = d(\nu_0 - M_{\text{P}}) - dM_{\text{P}} = d\nu_0 - 2dM_{\text{P}},$$

where we recall that d is the jumping rate between the two auxiliary regions and is linked to the diffusion constant, D, via equation (3). Solving this ODE gives

$$M_{\text{P}}(t) = \frac{1}{2}\left[\nu_0 - (\nu_0 - 2\mu_{\text{P}})e^{-2dt}\right]. \tag{31}$$

Assuming a small time-step, Δt, we can approximate $M_{\text{P}}(\Delta t)$, the number of particles after a time-step has occurred, by Taylor expanding equation (31) to first order:

$$\begin{aligned} M_{\text{P}}(\Delta t) &= M_{\text{P}}(0) + \Delta t M'_{\text{P}}(0) + o(\Delta t) \\ &\approx \frac{1}{2}\left[\nu_0 - (\nu_0 - 2\mu_{\text{P}})\right] + \frac{\Delta t}{2}\left[2d(\nu_0 - 2\mu_{\text{P}})\right] \\ &= (1 - 2d\Delta t)\mu_{\text{P}} + d\Delta t\nu_0. \end{aligned}$$

Fixing the value of D and using equation (3), we find that

$$M_{\text{P}}(\Delta t) \approx \left(1 - \frac{2D\Delta t}{h_a^2}\right)\mu_{\text{P}} + \frac{D\Delta t}{h_a^2}\nu_0.$$

We require the change in the number of particles over the small time-step to be small, and so would like $M_{\text{P}}(\Delta t) \approx \mu_P$. Thus we need to choose our parameters such that the quantity $D\Delta t/h_a^2$ small. This elucidates an important relationships between the fixed and free parameters of the model. If the diffusion coefficient is large then we must choose a small update time-step or a larger auxiliary region length to compensate.

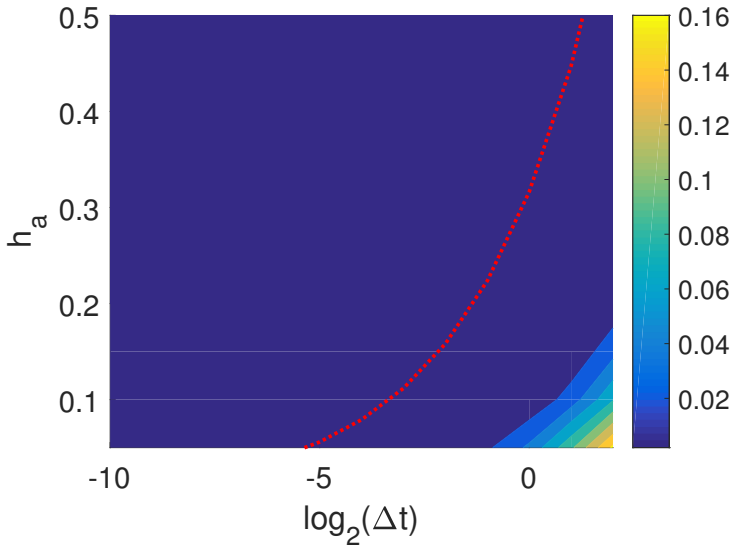


Figure 12. A contour plot for the HDE at time $T = 10$ for a series of simulations initialised with all particles uniformly distributed within the PDE region of the domain. Dark colours indicate low error. The red dashed line is the representative contour $D\Delta t/h_a^2 = 1/2$. Here, $D = 0.05$ and all simulations are averaged over 100 repeats.

Figure 12 shows that a large region of the $\Delta t - h_a$ space has a very low histogram distance error, meaning that our method is robust to parameter change, and only breaks down once the value of $D\Delta t/h_a^2$ becomes very large. The plot also shows that, given any choice of the width of the auxiliary regions, h_a, there is a value for the time-step, Δt, which will give a low level of error. Also, depending on our choice of Δt, we can adjust h_a to make the simulation more accurate.

6 Discussion

We have presented a new spatially coupled hybrid method for coupling a Brownian dynamics representation of a reaction-diffusion system to its corresponding mean-field PDE description. By bridging the gap in spatial scales with intermediate auxiliary regions, we have produced an algorithm that is not only accurate, but is also robust to the choice of the free parameters within the problem, namely the width of the auxiliary regions, h_a, and the fixed time-step, Δt used to update both the PDE and Brownian dynamics. This is in direct contrast to a previously presented PDE-to-Brownian hybrid, which we demonstrated to be extremely parameter-sensitive. In order to make the ARM even more robust, applicable and efficient, we now discuss several areas for possible extension, which will be addressed in future works.

In the interests of completeness we should point out that, as with the pseudo-compartment method of Yates and Flegg [49], the auxiliary region method requires that the mass in the PDE auxiliary region Ω_{PA} be sufficient for a step function, corresponding to the mass of a particle, to be removed uniformly from across the auxiliary region. This will lead to difficulties in situations in which particle numbers are low around the interface. Arguably though, we should not employ such hybrid methods in situations for which particle density is low around the interface as the PDE will be a poor model of the true stochastic, microscopic dynamics in these regions. A possible solution to this inconvenience, is the incorporation of an adaptive interface, which we have employed in test problem 4. Such interfaces evolve with the simulation dynamics, ensuring the appropriate model is used for the corresponding particle density [39].

A related issue is that of multiple interfaces. Multiple interfaces will allow the efficient simulation of stochastic reaction diffusion systems in which multiple regions of high and low concentration are expected. Such patterns will require interfaces to be dynamic in number and transient in nature. Although we have not implemented such interfaces in this work we expect it to be a relatively straightforward extension. While we have presented an example in which the system is simulated in a cuboid with a planar interface (test problem 4), non-planar interfaces, such as those which have corners or are curved, and complex domain geometries, present deeper challenges that we hope to address in a future publication.

Failing to maintain stochastic variation is a problem which is common amongst many spatially coupled hybrid methods. As a result of the deterministic nature of the PDE, the noise in the Brownian dynamics region of the domain is damped in comparison to the fully microscopic model (see Figure 6). In the literature, two approaches have been used in order to rectify this. The first is an overlap region, which has been employed in several papers [28, 21, 20]. These methods introduce a region of space which lies in the intersection of the two domains. In these regions, mass is simultaneously represented using both scales of description. The second is to replace the deterministic PDE with an appropriately chosen stochastic partial differential equation (SPDE). Alexander et al. [1] consider such a coupling and demonstrate they can indeed fix the discrepancy by using an SPDE as their continuum macro-scale model. We will address both the use of SPDEs and overlap regions (in which the region between the PDE and the Brownian dynamics regions is simulated using a purely compartment-based method) in forthcoming work.

The auxiliary region method provides a simple yet accurate method to couple an individual Brownian dynamics representation of a reaction-diffusion system to a corresponding PDE representation. Our hybrid algorithm will be of particular interest to researchers modelling reaction-diffusion systems whose concentrations vary significantly across the spatial domain. By reducing the computational expense of simulations, the ARM will facilitate the investigation of stochastic effects in such systems, in some cases, making the difference between being able to interrogate the system and not. In particular, we suggest that our method will be useful for the investigation of stochastic Turing patterns [16], Fisher waves [7, 6], oscillatory dynamics [29] and excitatory dynamics [22] with applications in embryogenesis [36], intracellular dynamics [32] and pattern formation [16] amongst others. It may also be worthwhile to interface the methods presented here with commonly used Brownian dynamics simulation software packages such as Smoldyn [3].

Data availability

The accompanying code has been uploaded as part of the supplementary material.

Competing interests

We have no competing interests.

Author's contributions

CAS and CAY contributed equally to the production of this manuscript. CAS performed the simulations and created the figures.

Funding

CAS is supported by a scholarship from the EPSRC Centre for Doctoral Training in Statistical Applied Mathematics at Bath (SAMBa), under the project EP/L015684/1.

Research ethics

An ethical assessment was not required prior to conducting this research.

Animal ethics

An ethical assessment was not required prior to conducting this research.

Permission to carry out fieldwork

No permissions were required prior to conducting this research.

Acknowledgements

CAY would like to thank the CMB/CNCB preprint club for constructive and helpful comments on a preprint of this paper.

Appendices

A Comparing to PDE-assisted Brownian dynamics

Within this section, we apply the same parameter values as used in Section **??** in order to demonstrate that the ARM can accurately simulate the problem that PDE assisted Brownian dynamics could not (see Figure 3). Recall, that we use $\Omega_{\text{P}} = (-1, 0)$, $\Omega_{\text{B}} = (0, 1)$ with the interface placed at $I = 0$. The only additional parameter that is to be defined is the auxiliary region width, which we set here to be $h_a = 0.1$. The results can be seen in Figure 13.

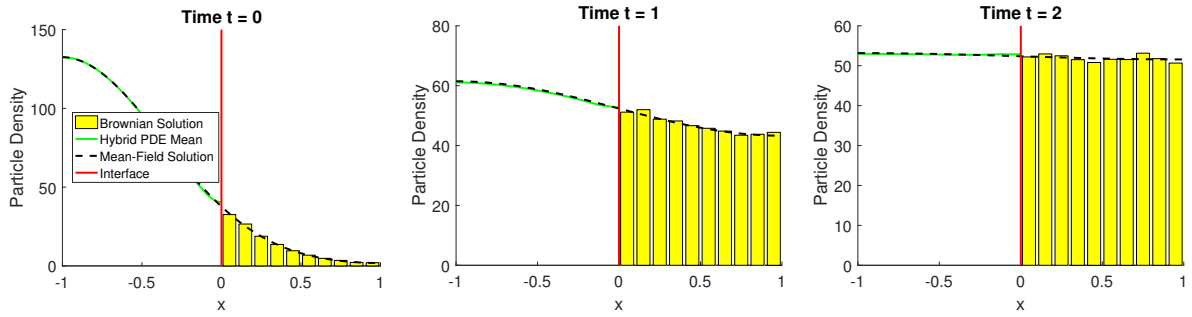


Figure 13. The evolution of the system corresponding to the reproduced figure from Franz et al. [21] (Figure 3), simulated using the ARM, at times (a) 0.2, (b) 1 and (c) 2. The colours and parameters are the same as in Figure 3, with the auxiliary region size being $h_a = 0.1$.

As can be seen from this figure, the agreement between the mean-field and hybrid solutions is much closer than that of the PDE assisted Brownian dynamics [21]. This indicates an improvement over the previous method. We also present the error plots which are described in Section **??** — namely the relative errors in particle numbers and the histogram distance errors.

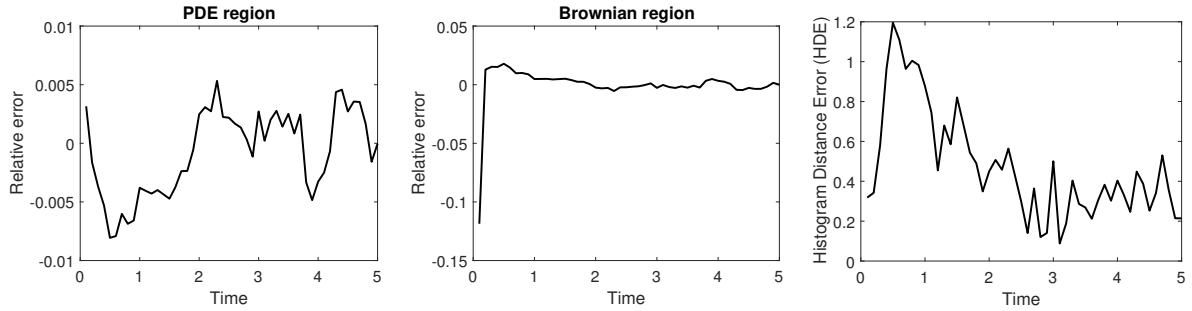


Figure 14. The error plots for the example in Figure 13. The first two panels show relative errors in particle numbers in the (a) PDE and (b) Brownian dynamics subdomains. The histogram distance error is displayed in (c).

Once again, the relative error plots (Figures 14(a)-(b)) appear to show no long-term bias in either direction and the histogram distance error (Figure 14(c)) is small.

References

1. F.J. Alexander, A.L. Garcia, and D.M. Tartakovsky. Algorithm refinement for stochastic partial differential equations: I. Linear diffusion. *J. Comput. Phys.*, 182(1):47–66, 2002.

2. V. Andasari, R.T. Roper, M.H. Swat, and M.A.J. Chaplain. Integrating intracellular dynamics using compucell3d and bionetsolver: applications to multiscale modelling of cancer cell growth and invasion. *PLoS One*, 7(3):e33726, 2012.

3. S.A. Andrews and D. Bray. Stochastic simulation of chemical reactions with spatial resolution and single molecule detail. *Phys. Biol.*, 1(3-4):137–151, 2004.

4. A.J. Black and A.J. McKane. Stochastic formulation of ecological models and their applications. *Trends Ecol. Evol.*, 27(6):337–345, 2012.

5. S.C. Brenner and C. Carstensen. *Finite element methods*, chapter 1. Encyclopedia of Computational Mechanics. John Wiley & Sons, Ltd, 2004.

6. H.-P. Breuer, W. Huber, and F. Petruccione. Fluctuation effects on wave propagation in a reaction-diffusion process. *Physica D*, 73(3):259–273, 1994.

7. H.-P. Breuer, W. Huber, and F. Petruccione. The macroscopic limit in a stochastic reaction-diffusion process. *Europhys. Lett.*, 30(2):69, 1995.

8. U. Dobramysl, S. Rüdiger, and R. Erban. Particle-based multiscale modeling of intracellular calcium dynamics. *Multiscale. Model. Sim.*, 14(3):997–1016, 2015.

9. M. Doi. Stochastic theory of diffusion-controlled reaction. *J. Phys. A.-Math. Gen.*, 9(9):1479, 1976.

10. D. Dürr, S. Goldstein, and J.L. Lebowitz. A mechanical model of brownian motion. *Commun. Math. Phys.*, 78(4):507–530, 1981.

11. J. Elf and M. Ehrenberg. Spontaneous separation of bi-stable biochemical systems into spatial domains of opposite phases. *Syst. Biol.*, 1(2):230–236, 2004.

12. T.J. English and D.A. Hammer. Brownian adhesive dynamics (BRAD) for simulating the receptor-mediated binding of viruses. *Biophys. J.*, 86(6):3359–3372, 2004.

13. R. Erban and S.J. Chapman. Stochastic modelling of reaction–diffusion processes: algorithms for bimolecular reactions. *Phys. Biol.*, 6(4):1–18, 2009.

14. R. Erban, M.B. Flegg, and G.A. Papoian. Multiscale stochastic reaction–diffusion modeling: application to actin dynamics in filopodia. *Bull. Math. Biol.*, 76(4):799–818, 2014.

15. R. Eymard, T. Gallouët, and R. Herbin. Finite volume methods. *Handbook of numerical analysis*, 7:713–1018, 2000.

16. M.B. Flegg. Smoluchowski reaction kinetics for reactions of any order. *SIAM J. Appl. Math.*, 76 (4):1403–1432, 2016.

17. M.B. Flegg, S.J. Chapman, and R. Erban. The two-regime method for optimizing stochastic reaction–diffusion simulations. *J. Roy. Soc. Interface*, 9(70):859–868, 2012.

18. M.B. Flegg, S.J. Chapman, L. Zheng, and R. Erban. Analysis of the two-regime method on square meshes. *(SIAM) J. Sci. Comput.*, 36(3):B561–B588, 2014.

19. M.B. Flegg, S. Hellander, and R. Erban. Convergence of methods for coupling of microscopic and mesoscopic reaction-diffusion simulations. *J. Comput. Phys.*, 289(C):1–17, 2015.

20. E.G. Flekkøy, J. Feder, and G. Wagner. Coupling particles and fields in a diffusive hybrid model. *Phys. Rev. E*, 64(6):066302, 2001.

21. B. Franz, M.B. Flegg, S.J. Chapman, and R. Erban. Multiscale reaction-diffusion algorithms: PDE-assisted Brownian dynamics. *SIAM J. Appl. Math.*, 73(3):1224–1247, 2013.

22. G. Gerisch, M. Ecke, R. Neujahr, J. Prassler, A. Stengl, M. Hoffmann, U.S. Schwarz, and E. Neumann. Membrane and actin reorganization in electropulse-induced cell fusion. *J. Cell Sci.*, 126(9): 2069–2078, 2013.

23. M.A. Gibson and J. Bruck. Efficient exact stochastic simulation of chemical systems with many species and many channels. *J. Phys. Chem. A.*, 104(9):1876–1889, 2000.

24. D.T. Gillespie. Exact stochastic simulation of coupled chemical reactions. *J. Phys. Chem.*, 81(25): 2340–2361, 1977.

25. D.T. Gillespie. Approximate accelerated stochastic simulation of chemically reacting systems. *J. Chem. Phys.*, 115(4):1716–1733, 2001.

26. D.T. Gillespie, A. Hellander, and L.R. Petzold. Perspective: Stochastic algorithms for chemical kinetics. *J. Chem. Phys.*, 138(17):170901, 2013.

27. A. Goriely and M. Tabor. Mathematical modeling of hyphal tip growth. *Fungal Biol. Rev.*, 22(2): 77–83, 2008.

28. J.U. Harrison and C.A. Yates. A hybrid algorithm for coupling PDE and compartment-based dynamics. *J. Roy. Soc. Interface*, 13(122):20160335, 2016.

29. M. Hoffmann and U.S. Schwarz. Oscillations of min-proteins in micropatterned environments: a three-dimensional particle-based stochastic simulation approach. *Soft Matter*, 10(14):2388–2396, 2014.

30. R. Holley. The motion of a heavy particle in an infinite one dimensional gas of hard spheres. *Probab. Theory. Rel.*, 17(3):181–219, 1971.

31. S.A. Isaacson. A convergent reaction-diffusion master equation. *J. Chem. Phys.*, 139(5):054101, 2013.

32. S. Khan, Y. Zou, A. Amjad, A. Gardezi, C.L. Smith, C. Winters, and T.S. Reese. Sequestration of CaMKII in dendritic spines *in silico*. *J. Comput. Neurosci.*, 31(3):581–594, 2011.

33. J. Lipková, K.C. Zygalakis, S.J. Chapman, and R. Erban. Analysis of brownian dynamics simulations of reversible bimolecular reactions. *SIAM J. Appl. Math.*, 71(3):714–730, 2011.

34. N.I. Markevich, J.B. Hoek, and B.N. Kholodenko. Signaling switches and bistability arising from multisite phosphorylation in protein kinase cascades. *J. Cell Biol.*, 164(3):353–359, 2004.

35. E. Moro. Hybrid method for simulating front propagation in reaction-diffusion systems. *Phys. Rev. E*, 69(6):060101, 2004.

36. R.L. Mort, R.J.H. Ross, K.J. Hainey, O.J. Harrison, M.A. Keighren, G. Landini, R.E. Baker, K.J. Painter, I.J. Jackson, and C.A. Yates. Reconciling diverse mammalian pigmentation patterns with a fundamental mathematical model. *Nat. Commun.*, 7:10288, 2016.

37. K.W. Morton and D.F. Mayers. *Numerical Solution of Partial Differential Equations*. Cambridge University Press, 2005.

38. G. Moy, B. Corry, S. Kuyucak, and S.H. Chung. Tests of continuum theories as models of ion channels. I. Poisson- Boltzmann theory versus Brownian dynamics. *Biophys. J.*, 78(5):2349–2363, 2000.

39. M. Robinson, M. Flegg, and R. Erban. Adaptive two-regime method: application to front propagation. *J. Chem. Phys.*, 140(12):124109, 2014.

40. M. Robinson, C. Luo, P.E. Farrell, R. Erban, and A. Majumdar. From molecular to continuum modelling of bistable liquid crystal devices. *Liq. Cryst.*, pages 1–18, 2017.

41. J.A. Sherratt. An analysis of vegetation stripe formation in semi-arid landscapes. *J. Math. Biol.*, 51(2):183–197, 2005.

42. C.A. Smith and C.A. Yates. Spatially extended hybrid methods: a review. *J. Roy. Soc. Interface*, 15(139), 2018.

43. G.D. Smith. *Numerical solution of partial differential equations: finite difference methods*. Oxford University Press, 1985.

44. M. Smoluchowski. Versuch einer mathematischen theorie der koagulationskinetik kolloider lösungen. *Z. Phys. Chem.*, 92(129-168):9, 1917.

45. A.M. Turing. The chemical basis of morphogenesis. *Phil. Trans. R. Soc. B.*, 237(641):37–72, 1952.

46. J.S. van Zon and P.R. ten Wolde. Green's-function reaction dynamics: a particle-based approach for simulating biochemical networks in time and space. *J. Chem. Phys.*, 123(23):234910, 2005.

47. V. Volpert and S. Petrovskii. Reaction–diffusion waves in biology. *Phys. Life Rev.*, 6(4):267–310, 2009.

48. G. Wagner and E.G. Flekkøy. Hybrid computations with flux exchange. *Phil. Trans. R. Soc. A*, 362:1655–1666, 2004.

49. C.A. Yates and M.B. Flegg. The pseudo-compartment method for coupling partial differential equation and compartment-based models of diffusion. *J. Roy. Soc. Interface*, 12(106):20150141, 2015.

50. R. ZhuGe, K.E. Fogarty, R.A. Tuft, L.M. Lifshitz, K. Sayar, and J.V. Walsh. Dynamics of signaling between ca^{2+} sparks and ca^{2+}-activated k^{+} channels studied with a novel image-based method for direct intracellular measurement of ryanodine receptor ca^{2+} current. *J. Gen. Physiol.*, 116(6): 845–864, 2000.

3.2 Conclusions

This chapter presents a novel spatially extended hybrid method, which couples the macroscopic regime to the microscopic. The main objective of this paper was to create a compartment-based coupling method which is accurate, efficient and robust to changes in parameter regimes. We have shown that this is possible, and that the resulting method is able to not only accurately simulate simple diffusion processes, but also those with second-order reactions and in three dimensions, highlighting its potential applicability to realistic biological applications.

In the wider context of the community, both mathematical and biological, this paper provides an important tool for modelling biological scenarios which have large spatial variation in particle numbers (for example the movement of a travelling wave (Moro, 2004)), or for those that require individual level detail in a region of space, but not elsewhere, such as the analysis of calcium dynamics in an intracellular environment (Dobramysl et al., 2015). We envisage that this method, with its simple coupling mechanism, will be of interest to anyone wishing to model a reaction-diffusion system both efficiently, but with the requisite accuracy that complex applications need.

The main open question that has come from this paper is how such methods are applicable to a growing domain. The addition of a growing domain will allow hybrid methods to be utilised in more biological contexts. For example, when considering embryonic growth, the movement and proliferation of neural crest cells occur on a growing domain, and long-distance cell migration has been shown to be an important mechanism (McLennan et al., 2012). In the paper of McLennan et al. (2012), they utilise a hybrid method that grows in space in order to investigate this movement and proliferation. However, the hybrid method that they use is a so-called "species splitting" hybrid method. That is, the chemoattractant is modelled using a PDE, while the neural crest cells use a microscopic representation, with an interaction between the two. We wish to create spatially extended methods, where the subdomains are distinct, which is the focus of Chapter 6.

Related to the addition of the growing domain, we also require some of the test problems that we have used in this chapter to also be modelled on a growing domain. In Chapter 5, we generate an equivalence framework between the three methods for a generic family of boundary conditions, in which the formation of a morphogen gradient (Section 4.3) is part of, on a growing domain. This will then feed directly into the test problems for the paper of Chapter 6.

Chapter 4

Unbiased on-lattice domain growth

This chapter is made up of a paper published in *Physical Review E* in 2019, authored by myself, Cécile Mailler and Kit Yates. This paper is the first of two (together with Chapter 5) which establish some of the equivalence frameworks required for the implementation of hybrid methods on a growing domain. Specifically, in this paper we develop a method for growing the mesoscopic representation of reaction-diffusion processes, particularly when in a low diffusion regime. A low diffusion regime is classed as one in which the particles diffuse at a much slower rate than they are dispersed by the domain.

The main motivation behind this paper was to create an alternative domain growth method for the mesoscale that firstly stretches uniformly, meaning that particle mass spreads evenly across the entire domain through time, and also performs well under all parameter regimes. This is in contrast to a method (that is well utilised in the literature) devised by Baker et al. (2010). As part of this method, coined the "original method" in the paper presented in this chapter, a compartment is chosen to grow at random. It doubles in length and simultaneously splits in two. The compartments to the right of the one chosen to split are pushed to the right, and the particles in the chosen compartment are redistributed using a binomial random variable. With large growth rates compared to diffusion, we have demonstrated that this method does not maintain the expected uniform particle profile. In contrast, our method, named the stretching method, redistributes particles in all compartments using "overlap" regions. The stretching method is able to perform under all parameter regimes.

This chapter demonstrates the equivalence between the macroscale and the mesoscale models of diffusion required in order to create one of the three hybrid methods in Chapter 6. This equivalence compares the stretching method to the diffusion PDE on an exponentially growing domain, which is obtained by taking the diffusive limit of the mean equations, which we calculate in Appendix D of the paper.

In this paper, we do not test the method on any examples which contain reactions of second or higher order. We do this in order to remove complexity so that any errors that we find between the stretching method and its associated mean-field PDE can be attributed to the method that we develop and not to disagreements between the PDE and the mean of the stochastic process. Under higher-order reactions, in the small compartment limit, the RDME breaks down, and a convergent reaction diffusion master equation is instead required (Isaacson, 2013). We anticipate that the algorithm would still be appropriate in its current form, however this remains to be tested. We also note that we assume that there is no volume exclusion in our model, which would also alter the dynamics of the RDME (Cianci et al., 2017), and is once again an avenue for future work.

Since the mesoscale is an important component of all three of the methods described in Chapter 6 (as it is the only mechanism by which particles may move between the two subdomains for each of the hybrid methods), it is of great importance to have an accurate modelling methodology at this scale. While the stretching method is of limited use for the extension of the ARM (Chapter 3), it will become very important for the extension of both the pseudo-compartment method (Yates and Flegg, 2015) and the ghost cell method (Flegg et al., 2015).

4.1 Overview of the paper

This paper begins with an introduction to mesoscopic modelling and a qualitative description of the shortcomings of the method described by Baker et al. (2010). In Section 2, we conduct both a mathematical and numerical analysis of this previous method, demonstrating that it fails to match the mean-field diffusion PDE when diffusion is much smaller than domain growth. Section 3 contains an explanation of the new stretching method, together with an algorithm for its implementation, together with a pair of test problems to assess its performance. This section also contains a comparison of the two methods, with an investigation into when the method of Baker et al. (2010) has an "acceptable" level of error, comparable with the stretching method. Finally, the

implications of the paper and possible next steps are discussed in Section 4.

Appendix 6B: Statement of Authorship

This declaration concerns the article entitled:			
Unbiased on-lattice domain growth			
Publication status (tick one) **Draft manuscript** ☐ **Submitted** ☐ **In review** ☐ **Accepted** ☐ **Published** ●			
Publication details (reference)	Journal – Physical Review E, 100 (6), 063307 Authors – Cameron A. Smith, Cécile Mailler, Christian A. Yates		
Copyright status (tick the appropriate statement) I hold the copyright for this material ● Copyright is retained by the publisher, but I have been given permission to replicate the material here ☐			
Candidate's contribution to the paper (provide details, and also indicate as a percentage)	Most calculations have been performed by the author of this thesis (80%). All numerical computations and simulations have been completed by the author of this thesis (100%). All authors contributed equally to the presentation of the content (33%).		
Statement from Candidate	This paper reports on original research I conducted during the period of my Higher Degree by Research candidature.		
Signed		**Date**	04/06/2021

Unbiased on-lattice domain growth

Cameron A. Smith[1,*], Cécile Mailler[2], Christian A. Yates[1]
[1]**Centre for Mathematical Biology, Department of Mathematical Sciences, University of Bath, Claverton Down, Bath, BA2 7AY, United Kingdom**
[2]**Probability Laboratory, Department of Mathematical Sciences, University of Bath, Claverton Down, Bath, BA2 7AY, United Kingdom**
[*]**E-mail: c.smith3@bath.ac.uk**

Key words: domain growth, mesoscopic modelling, reaction-diffusion master equation, morphogen gradient, uniform growth, on-lattice modelling

Abstract

Domain growth is a key process in many areas of biology, including embryonic development, the growth of tissue, and limb regeneration. As a result, mechanisms for incorporating it into traditional models for cell movement, interaction, and proliferation are of great importance. A previously well-used method in order to incorporate domain growth into on-lattice reaction-diffusion models causes a build up of particles on the boundaries of the domain, which is particularly evident when diffusion is low in comparison to the rate of domain growth. Here, we present a new method which addresses this unphysical build up of particles at the boundaries, and demonstrate that it is accurate for scenarios in which the previous method fails. Further, we discuss for which parameter regimes it is feasible to continue using the original method due to diffusion dominating the domain growth mechanism.

1 Introduction

Domain growth is an inherent feature of many biological systems, from neural crest cell migration [1, 2] to the growth and shrinkage of tissue [3, 4], and it has been investigated in the context of pattern formation in reaction-diffusion systems [5, 6]. It is therefore important that we have reliable mathematical tools in order to model such systems.

One method of modelling general reaction-diffusion systems is to compartmentalise the spatial domain into a lattice of small regions, in which particles are considered to reside. Typically, particles are permitted to jump between neighbouring compartments (although non-local jumping can also be incorporated [7]), and particles may react with others in their current compartment (similarly, in some methods, particles may be allowed to interact with particles in neighbouring compartments [8]). These events, under the assumption that updates happen in continuous time, are considered Markovian and, as a result, have associated exponentially distributed waiting times. The most commonly used method to simulate such systems is the Gillespie algorithm [9], however there are many others that are also used within the literature (see, for example, the next reaction method [10], the next subvolume method [11, 12], and the sorting direct method [13]).

The word 'particle' in this context can refer to a multitude of biological or physical entities; a particle may be a cell when modelling biological systems such as neural crest cell migration in embryonic development, a chemical species when considering pattern formation or different classes of individual in the case of the spread of an epidemic. The discretisation of the domain does not necessarily correspond to any physical attribute of the space being modelled, but is usually used as a mathematical tool through which we are able to model stochasticity in particle positions and domain growth. As an example, consider the formation of a morphogen gradient on a growing domain [14]. In this example, the particles are the morphogen molecules, and the domain may be discretised to arbitrary accuracy, with no physical meaning necessarily being attached to the compartments. This case study will be investigated in more detail in Section 3.1.

One possible option for incorporating domain growth into these on-lattice position-jump processes has been previously suggested by Baker et al. [15]. Growth is achieved by choosing compartments to

divide uniformly at random at a given rate. Once chosen, a compartment instantaneously doubles in length and splits down the middle to produce a new compartment (pushing all compartments to the right of the one chosen by one compartment's width). Particles that resided in the original compartment are then redistributed into the newly created compartments by placing each particle into one or the other with equal probability. This is illustrated schematically in Figure 1, and a method for its implementation is detailed in Algorithm 1. A commonly implemented simulation allows particles to diffuse by jumping between neighbouring boxes while, at the same time, attempting to grow the domain uniformly [15, 3, 16]. When simulating this scenario using the method of Baker et al. [15] and averaging over multiple repeats, it becomes apparent that there is an issue with particles building up at the ends of the domain, an effect that can be expected when diffusion is low in comparison to the domain growth rate. When diffusion is larger, there is enough time for particles to relax to equilibrium (particles have enough time to occupy newly created sites) before the next domain growth event.

Algorithm 1: The original method [15]

(1a) Initialise time $t = 0$ and set the final time $T > 0$. Initialise the number of compartments k and the particle numbers in each compartment $m_i,\ i \in \{1, ..., k\}$. Specify the size of compartment h, the jump rate $d = D/h^2$ and the growth rate ρ.

(1b) Calculate the propensity functions $\alpha_i^L = dm_i$, $\alpha_i^R = dm_i$ and $\alpha_i^G = \rho$ for each compartment $i \in \{1, ..., k\}$ for left jumps, right jumps and growth events respectively, with $\alpha_1^L = 0 = \alpha_k^R$. Calculate the sum of the propensity functions:

$$\alpha_0 = \sum_{i=1}^{k} \left(\alpha_i^L + \alpha_i^R + \alpha_i^G\right).$$

(1c) Calculate the time until the next event by firstly drawing a uniform random variable between zero and one, $u_1 \sim \text{Unif}(0, 1)$, and setting $\tau = 1/\alpha_0 \log(1/u_1)$. Update the time $t \leftarrow t + \tau$.

(1d) Determine which event, amongst all compartments, is next to occur by choosing one at random with probability proportional to the propensity function.

1. If the event corresponds to a left (resp. right) jump event from compartment i, remove a particle from compartment i and add one to compartment $i - 1$ (resp. $i + 1$).
2. If the event corresponds to a growth event in compartment i: Record the pregrowth particle numbers $\boldsymbol{r} \leftarrow \boldsymbol{m}$, create an extra compartment at the right end of the postgrowth domain (increasing k by 1 by setting $k \leftarrow k+1$), draw a binomial random variable with r_i trials and probability of success $1/2$, $b \sim \text{Bin}(r_i, 1/2)$, and redistribute particles as follows (for $j \in \{1, ..., k\}$):

$$m_j = \begin{cases} r_j, & \text{if } j \in \{1, ..., i-1\}, \\ b, & \text{if } j = i, \\ r_i - b, & \text{if } j = i+1, \\ r_{j-1}, & \text{if } j \in \{i+2, ..., k\}. \end{cases}$$

(1e) If $t < T$ then return to step **(1b)**, else end.

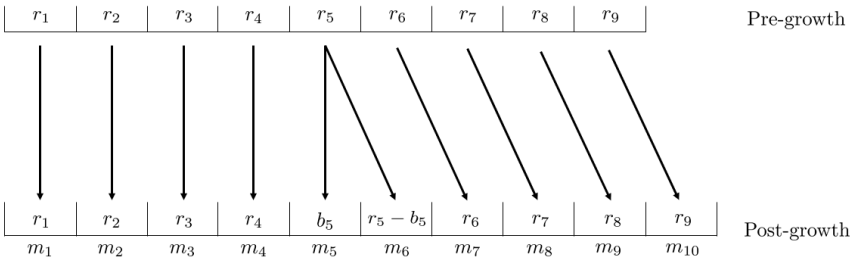


Figure 1. A schematic showing a possible splitting event when there are nine pregrowth compartments (compartments identified by the subscripts on the particle numbers). Pregrowth particle numbers are denoted r_i. In this case, compartment 5 is chosen to divide, and its contents are split binomially with probability of success 1/2 ($b_5 \sim \text{Bin}(r_5, 1/2)$) between the compartment in the original position (compartment 5 in this case) and the new one that is created to its right. The compartments originally numbered 6, 7, 8 and 9 are moved one position to the right and become compartments 7, 8, 9 and 10, respectively. Postgrowth particle numbers are denoted by the m_i's.

In order to rectify this problem we have designed a novel mechanism which enables the implementation of unbiased domain growth — one that prevents the accumulation of particles at the boundaries. This method enacts a more continuous approach to domain growth compared to Baker et al. [15], in which compartments are first stretched and then renormalised with concurrent redistribution of their particles in to neighbouring compartments. In general, we will consider growth dynamics in isolation, without considering any other mechanisms such as diffusion and reactions. This is in order to demonstrate the possible issues without confounding or hiding growth induced phenomena with other dynamics. All models will be presented in one dimension, but extensions to higher dimensions will be discussed in Section 4.

This paper will be set out in the following way. In Section 2 we investigate and explain the problems with the commonly employed domain growth mechanism in more detail. In Section 3, we present a novel, but distinct and importantly unbiased domain growth method that we use to address this issue. Finally, we present our conclusions in Section 4.

2 Problems with an existing domain growth mechanism

Within this section, we describe and demonstrate problems with an existing domain growth mechanism at the mesoscale [15], which we will refer to as the "original method". We then postulate why the phenomenon has not been noticed before and confirm the existence of the problem using three different techniques.

The original method causes an accumulation of particles at the ends of the domain. This build up is caused by the inherent bias in the way in which growth is implemented: compartments are always shifted to the right when one is chosen to split. When a single compartment is chosen to split, most postgrowth compartments will either retain their pregrowth contents (if a pregrowth compartment to its right splits) or will take on the contents of the pregrowth compartment to its left (if the split is to the left of that pregrowth compartment). The notable exceptions to these rules are when postgrowth compartments retain or gain (respectively) only half of the particles (on average) from its or a neighbour's (respectively) pregrowth compartment. These events only affect two compartments per splitting event, but are nevertheless important.

The only way a postgrowth compartment retains half of its particles (on average) is if it is the compartment chosen to split. Similarly, the only way a postgrowth compartment gains half of the particles (on average) of a neighbouring compartments is if the pregrowth neighbour immediately to the left splits. All the postgrowth compartments, therefore, gain or retain half a compartments worth of particles (on average) in two different ways. The only exceptions are the first and last postgrowth compartments. The first postgrowth compartment can only retain half of its particles (on average) if it is chosen to split. It can never gain half of the particles from another compartment as there are no pregrowth compartments to its left that can split. Similarly, the last postgrowth compartment can never retain half of its particles since it did not exist on the pregrowth domain. Instead it can only gain half of the particles from a neighbouring compartment when the pregrowth compartment in the final position splits.

If particles are initially distributed uniformly, then an unbiased domain growth method will maintain this uniform distribution (on average, and provided no particles enter or leave the domain). For the method of Baker et al. [15] (see Figure 1 and Algorithm 1), splitting events will redistribute the particles into the postgrowth compartments which correspond to the pregrowth compartment chosen to split and its neighbour to the right. For example, in Figure 1, the particle redistribution event affects postgrowth compartments 5 and 6 when pregrowth compartment 5 is chosen to split. If these splitting events affected all compartments equally then the domain growth method would be unbiased and the particle profile would remain uniform as the domain grew. However, since the two end compartments suffer (on average) only half of the particle-reducing splitting events that the non-end compartments suffer, this leads to particles accumulating at the ends of the domains.

The build up of particles at either end of the domain has previously been hidden by the smoothing effect of fast diffusion [15]. If diffusion is large in comparison to the rate of domain growth, particles are able to diffuse away from the high concentration regions at the end of the domain, leading to a near-uniform particle profile (given a uniform initial condition and zero-flux boundary conditions).

We will illustrate the problems with the compartment-based domain growth method of Baker et al. [15] in three different ways. For the first we undertake a stochastic simulation of the original method using the Gillespie algorithm [9] and average particle densities over many repeats. For the second demonstration, we calculate the numerical solution of the mean-field equations that stem from the corresponding domain growth master equation of the stochastic process in the absence of diffusion. Thirdly, we employ an analytical mathematical argument based on local redistribution of particles to assess the particle distribution in the limit of large numbers of boxes. All code written in order to simulate these, and all other examples within this manuscript, can be found in the electronic supplementary material [17].

For the stochastic simulation, we initialise a total of 1000 particles uniformly in the domain $[0, 4]$ which grows exponentially in time with rate ρ (we will use exponential growth as our primary test simulation, however other types of growth exhibit similar problems). We set the compartment width to be 0.4 so that initially we have 10 compartments. The growth rate, ρ, is set to be 0.01 (note that all units here are arbitrary in both space and time). We choose compartment splitting times to be deterministic, meaning that we pre-calculate the times at which a compartment should split (on average) and always enact a splitting event at those times, while the Gillespie algorithm handles the stochastic events between these pre-determined times. Alternatively, growth events could be incorporated stochastically as part of the Gillespie algorithm, but for the purposes of demonstrating the problem, and for ease of visualisation, we use deterministic growth in this exposition. The compartment to divide at each predetermined splitting time is chosen uniformly at random from amongst the current compartments. Diffusion is set to be 0, so that the only effect on compartment occupancy is domain growth. Using this domain growth configuration and averaging over 50,000 repeats, we see the clear build up of particles at both ends of the domain (see Figure 2).

To further corroborate these results and explain where this behaviour originates from, we consider the master equation for the splitting algorithm outlined above, which was first derived in [15]. Let

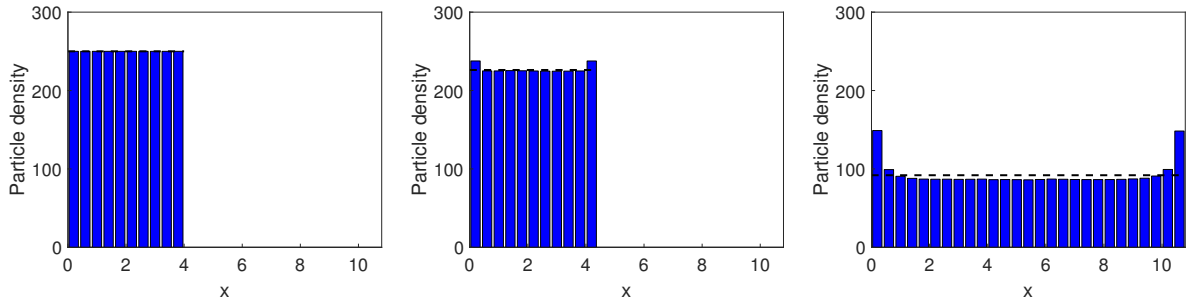


Figure 2. Three snapshots of the stochastic simulation which demonstrate the issue with the original domain growth method [15]. The blue bars denote particle densities in each compartment, and the black-dashed lines denote the expected behaviour of an unbiased growth method. Particle numbers are averaged over 50,000 repeats. All other parameter values are as in the text. (a) The initial configuration shows 1,000 particles spread uniformly across the initial domain $[0, 4]$. (b) At time 10, particle build up at the domain ends is already evident. (c) Once the system has evolved to time 100, particles have accumulated significantly close to the domain ends.

$p(\boldsymbol{m}, k, t) = \mathbb{P}(\boldsymbol{N}(t) = \boldsymbol{m}, K(t) = k)$ be the probability that the state variable (the number of particles in each compartment) at time t is $\boldsymbol{m} = (m_1, ..., m_k)^T$ and the number of compartments at time t is k. This quantity evolves according to:

$$\frac{\mathrm{d}p}{\mathrm{d}t}(\boldsymbol{m}, k, t) = \rho \sum_{j=1}^{k-1} \pi(m_j, m_{j+1}|m_j + m_{j+1}) p(G_j \boldsymbol{m}, k-1, t) - \rho k p(\boldsymbol{m}, k, t). \tag{1}$$

Here ρ is the splitting rate, which is the rate at which each compartment is chosen to divide, $\pi(x, y|z)$ is a distribution describing the probability that, given there are z particles in a compartment before splitting, there are x and y particles in the two post-split compartments (where $x + y = z$), and $G_j : \mathbb{R}^k \to \mathbb{R}^{k-1}$ is an operator that combines the contents of compartments j and $j + 1$ (the opposite process to splitting), so that

$$G_j : (m_1, ..., m_j, m_{j+1}, ..., m_k)^T \mapsto (m_1, ..., m_j + m_{j+1}, ..., m_k)^T.$$

Note that the growth considered for the master equation is stochastic in both position and timing.

We can calculate the mean-field equations for the evolution of the mean number of particles in each compartment, under the splitting probability corresponding to a symmetric binomial distribution

$$\pi(x, y|z) = \binom{z}{x} \left(\frac{1}{2}\right)^z,$$

by multiplying both sides of equation (1) by m_i for each index $i \in \{1, ..., k\}$, and summing over the entire postgrowth state variable. Define the quantity

$$M_i^k(t) = \sum_{\boldsymbol{m} \in \mathbb{N}_0^k} m_i p(\boldsymbol{m}, k, t),$$

which is the mean number of particles in compartment i at time t, given that there are k compartments in the system overall. These corresponding mean-field equations are given by:

$$\frac{\mathrm{d}M_i^k}{\mathrm{d}t}(t) = \rho \left[\left(i - \frac{3}{2}\right) M_{i-1}^{k-1}(t) + \left(k - \frac{1}{2} - i\right) M_i^{k-1}(t) - k M_i^k(t) \right], \tag{2}$$

which holds for $k \in \mathbb{N}$, $j \in \{1, ..., k\}$ and $t > 0$ [15], noting that $M_k^{k-1} \equiv 0$. The full derivation is omitted here, however we direct the interested reader to Baker et al. [15], or Appendix D.1 for a similar calculation. We plot the solutions to equation (2) in order to illustrate the average behaviour of the system under the biased domain growth algorithm, in Figure 3. We show that in this case, we exhibit the same build up of particles at the boundaries that is evident in the stochastic simulation, shown in Figure 2.

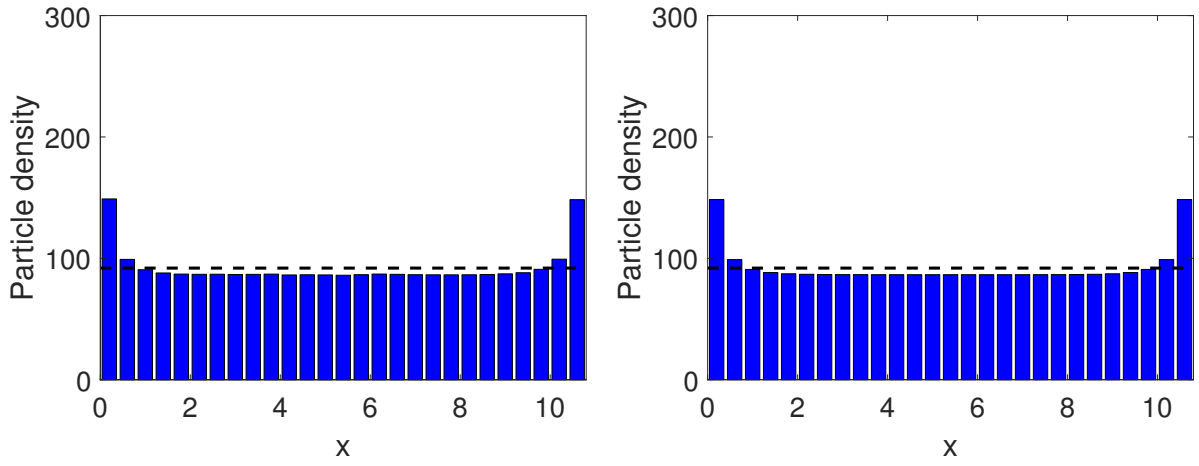


Figure 3. A comparison between (a) the particle density found by using Algorithm 1, averaged over 50,000 repeats (reproduced from Figure 2(c)), and (b) the corresponding particle density found by solving mean-field equations (2). The mean-field equations are initialised in the same way as the stochastic algorithm (see Figure 2(a) and related caption).

Finally, we gain a more quantitative insight into the bias engendered by the domain growth method of Baker et al. [15] by using an analytical approach. In particular, we derive coefficients which describe the densities of particles in each compartment. We let u_i^k denote the normalised density of particles in compartment i when there are k compartments in total (a random variable), $M_i^k = \mathbb{E}[u_i^k]$, and I^k is the index of the compartment which splits when there are k compartments in total (chosen uniformly at random for each event). Finally, we define q to be a sample from some symmetric distribution with mean $1/2$. This will denote the random proportion of density that is placed in the left-hand postgrowth compartment when there are k compartments in total, and is independent of the particle density and the compartment chosen to split. We begin with all of the density in the first (and only) compartment, so that $u_1^1 = 1$. We will set up recursion relationships between the u_i^k's for different i and k in order to approximate u_i^k for large k. Specifically, we express u_i^{k+1} in terms of u_j^k for $j \in \{1, ..., k\}$:

$$u_i^{k+1} = \underbrace{u_i^k \mathbb{1}_{[I^k > i]}}_{\substack{\text{Compartment} \\ \text{to right of} \\ i \text{ splits}}} + \underbrace{q u_i^k \mathbb{1}_{[I^k = i]}}_{\substack{\text{Compartment} \\ i \text{ splits}}} + \underbrace{(1-q) u_{i-1}^k \mathbb{1}_{[I^k = i-1]}}_{\substack{\text{Compartment} \\ i-1 \text{ splits}}} + \underbrace{u_{i-1}^k \mathbb{1}_{[I^k < i-1]}}_{\substack{\text{Compartment} \\ \text{to left of} \\ i-1 \text{ splits}}}, \qquad i \in \{2, ..., k\}, \tag{3}$$

with similar expressions for $i = 1$ and $k + 1$. Here $\mathbb{1}_{[\text{condition}]}$ is the indicator function, which is unity when the subscripted condition is satisfied and zero otherwise.

Considering the first compartment, relationship (3) stipulates that

$$u_1^{k+1} = u_1^k \mathbb{1}_{[I^k > 1]} + q u_1^k \mathbb{1}_{[I^k = 1]}.$$

Taking expectations of this expression, noting that the choice of compartment, the density in each compartment and the proportion of the density placed in the left pregrowth compartment are independent (so that u_i^k, I^k and q are independent for every $i \in \{1, ..., k\}$) we find that:

$$\begin{aligned} M_1^{k+1} &= M_1^k \mathbb{P}(I^k > 1) + \frac{1}{2} M_1^k \mathbb{P}(I^k = 1), \\ &= M_1^k \left(\frac{k-1}{k} + \frac{1/2}{k} \right), \\ &= M_1^k \left(1 - \frac{1/2}{k} \right). \end{aligned}$$

Applying this recursive relation k times:

$$M_1^{k+1} = M_1^1 \prod_{n=1}^{k} \left(1 - \frac{1/2}{n} \right) = \prod_{n=1}^{k} \left(1 - \frac{1/2}{n} \right), \tag{4}$$

since $M_1^1 = \mathbb{E}[u_1^1] = 1$. In order to simplify this expression, we can use the following relationship, which is derived in Appendix A:

$$\prod_{n=a}^{b} \left(1 + \frac{c}{n} \right) \approx \frac{\Gamma(a)}{\Gamma(a+c) e^c} b^c, \tag{5}$$

where $a, b \in \mathbb{N}$ such that $a < b$ and b is large, and $c \in \mathbb{R}$. Applying approximation (5) to equation (4) for large k gives:

$$M_1^{k+1} = \prod_{n=1}^{k} \left(1 - \frac{1/2}{n} \right) \approx \frac{\Gamma(1)}{\Gamma(1/2) e^{-1/2}} k^{-1/2} = \frac{1}{\sqrt{\pi} e^{-1/2}} k^{-1/2}. \tag{6}$$

Now consider the second compartment. As with the first, we write the recursion relation (3):

$$u_2^{k+1} = u_2^k \mathbb{1}_{[I^k > 2]} + \frac{1}{2} u_2^k \mathbb{1}_{[I^k = 2]} + \frac{1}{2} u_1^k \mathbb{1}_{[I^k = 1]}.$$

Once again, we take expectations and simplify by applying relation (3) recursively (as in equation (4)):

$$\begin{aligned} M_2^{k+1} &= M_2^k \mathbb{P}(I^k > 2) + \frac{1}{2} M_2^k \mathbb{P}(I^k = 2) + \frac{1}{2} M_1^k \mathbb{P}(I^k = 1), \\ &\vdots \\ &= \underbrace{M_2^2 \prod_{n=2}^{k} \left(1 - \frac{3/2}{n} \right)}_{\text{Term 1}} + \underbrace{\sum_{n=2}^{k-1} \left(\prod_{j=n+1}^{k} \left(1 - \frac{3/2}{j} \right) \right) M_1^n \left(\frac{1/2}{n} \right)}_{\text{Term 2}} + \underbrace{M_1^k \left(\frac{1/2}{k} \right)}_{\text{Term 3}}. \end{aligned} \tag{7}$$

We consider each of the three terms in equation (7) sequentially. Using equation (5), we can approximate term 1, for large k, as

$$\prod_{n=2}^{k} \left(1 - \frac{3/2}{n} \right) \approx \frac{\Gamma(2)}{\Gamma(1/2) e^{-3/2}} k^{-3/2}. \tag{8}$$

It can be shown (see Appendix B) that for large k, term 2 can be approximated by

$$\sum_{n=2}^{k-1} \left(\prod_{j=n+1}^{k} \left(1 - \frac{3/2}{j} \right) \right) M_1^n \left(\frac{1/2}{n} \right) \approx \frac{1}{2} \frac{1}{\sqrt{\pi} e^{-1/2}} k^{-1/2} - \frac{1}{\sqrt{\pi} e^{-1/2}} k^{-3/2}. \tag{9}$$

To simplify term 3 we make use of result (6):

$$M_1^k\left(\frac{1/2}{k}\right) \approx \frac{1}{2}\frac{1}{\sqrt{\pi}e^{-1/2}}k^{-3/2} \tag{10}$$

Substituting the resultant expressions (8), (9) and (10) into (7) gives, for large k:

$$M_2^{k+1} \approx M_2^2\frac{\Gamma(2)}{\Gamma(1/2)e^{-3/2}}k^{-3/2} + \frac{1}{2}\frac{1}{\sqrt{\pi}e^{-1/2}}k^{-1/2} - \frac{1}{\sqrt{\pi}e^{-1/2}}k^{-3/2} + \frac{1}{2}\frac{1}{\sqrt{\pi}e^{-1/2}}k^{-3/2},$$

For large k, the $O(k^{-1/2})$ terms dominate leaving us with the following approximation:

$$M_2^{k+1} \approx \frac{1}{2}\frac{1}{\sqrt{\pi}e^{-1/2}}k^{-1/2}. \tag{11}$$

Following the same procedure, we can find the approximate expressions for the asymptotic particle density in each of the compartments. In particular, it can be shown that

$$M_i^k \approx c_i k^{-1/2}, \tag{12}$$

where

$$c_1 = \frac{1}{\sqrt{\pi}e^{-1/2}},$$
$$c_i = c_1\prod_{j=2}^{i}\frac{2j-3}{2j-2}, \quad i \in \{2,3,...\}.$$

To assess the accuracy of this approximation, we undertake a stochastic simulation initialised with a single compartment containing 100 particles. At time $t = 100$, under our time-deterministic splitting mechanism each simulation finishes with 20 compartments. We then compare the particle numbers in each compartment, averaged over 10,000 repeats, to equation (12). Since the simulation domain has only finitely many compartments but our mathematical analysis considers an infinite number of compartments, we average M_i^{20} and M_{21-i}^{20} when we plot compartment i under the assumption that densities are initialised, and subsequently remain, symmetric. Finally, we scale each of the plots so that they have the same number of particles. Although a quantitative agreement is not expected, since our results hold strictly only on an infinite domain, the results in Figure 4 demonstrate that our mathematical analysis matches the simulation results qualitatively.

The original domain growth method proposed by Baker et al. [15] has been used in many compartment-based studies of domain growth [18, 19, 20, 3]. Despite not having previously been evident, we have been able to demonstrate in three distinct ways, that this domain growth method is biased. The consequence of this bias is that particles tends to accumulate at the extreme ends of the domain. In the next section, we introduce the stretching method, which prevents this build up of particles and gives genuinely uniform, unbiased domain growth.

3 Stretching method

We now introduce the stretching method. This differs from the original method because it is a global method as opposed to a local one. That is, instead of choosing a single compartment to instantaneously grow to twice its length and split, we stretch all compartments by a small amount and redistribute particles amongst all compartments (for a brief discussion of local growth mechanisms, please see Appendix C). We will firstly explain the method, before demonstrating its effectiveness. We do this by showing that it

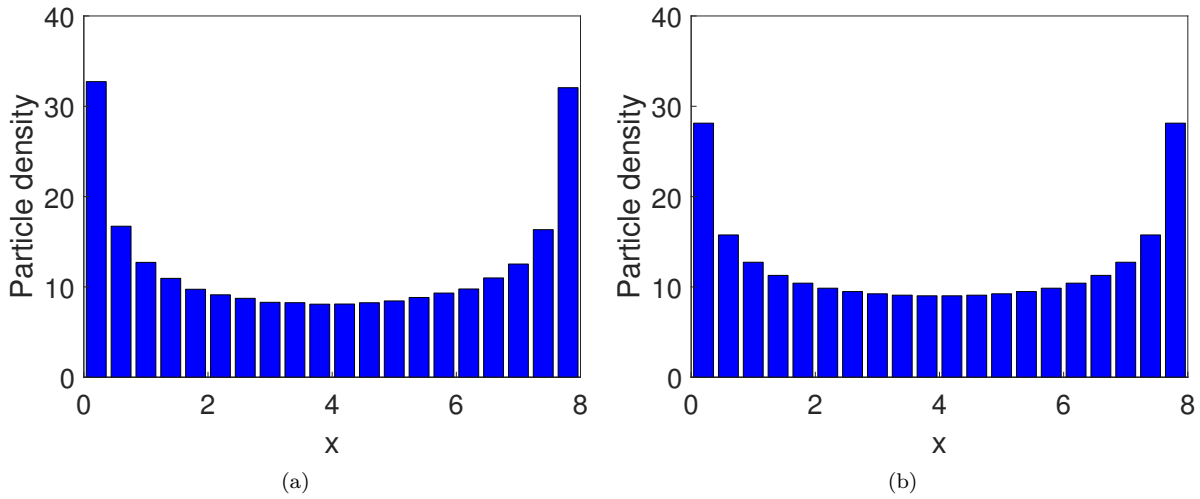


Figure 4. A comparison between (a) a stochastic simulation averaged over 10,000 repeats and initialised with a single compartment and (b) the result of the mathematical argument, where the displayed compartment i is given by the average of M_i^{20} and M_{21-i}^{20}, calculated using equation (12). Both plots have the same total area.

can correctly maintain a uniform particle profile on a uniformly growing domain. We then look at a case study, the formation of a morphogen gradient on an exponentially growing domain, in order to directly compare the original and stretching methods using an example with its roots in biology (see Section 3.1). Finally, in Section 3.2, we investigate the parameter regimes in which the spatial inhomogeneities in the original method are negligible.

We begin by describing the method. Assume the number of compartments before a growth event is $k-1$ for some $k > 2$, and define each compartment to be of width h (see Figure 5(a)). We will define the state variable before growth to be $\boldsymbol{r} \in \mathbb{N}_0^{k-1}$ and after growth to be $\boldsymbol{m} \in \mathbb{N}_0^k$ in order to be consistent with the original method. The method proceeds as follows:

1. When a growth event is chosen to occur, we stretch the domain to be of size kh rather than $(k-1)h$ (see Figure 5(b)). We do this uniformly across the entirety of the domain, so that each compartment on the stretched domain is now of width $kh/(k-1)$.

2. In the second step we add a compartment to the right-hand end of the pre-stretched domain (see Figure 5(c)). It is on this postgrowth domain that we define the state variable $\boldsymbol{m}$. Note that we now have two domains, each with a different number of compartments, but both of the same length.

3. For the third step, we compare the two meshes. Note that for every stretched compartment, exactly two of the postgrowth compartments intersect it (see Figure 5(d)). Assuming particles are uniformly distributed across each stretched compartment, we can calculate the proportion of particles, δ_i^{k-1}, that should be placed in the left overlapping compartment. If we denote the right-hand end of compartment i in the renormalised domain as $x_i = ih$ and use $\tilde{x}_i = [ki/(k-1)]\,h$ the quantity for the stretched domain, then:

$$\begin{aligned}\delta_i^{k-1}\frac{k}{k-1}h &= x_i - \tilde{x}_{i-1} \\ &= ih - \frac{k}{k-1}ih + \frac{k}{k-1}h\end{aligned}$$

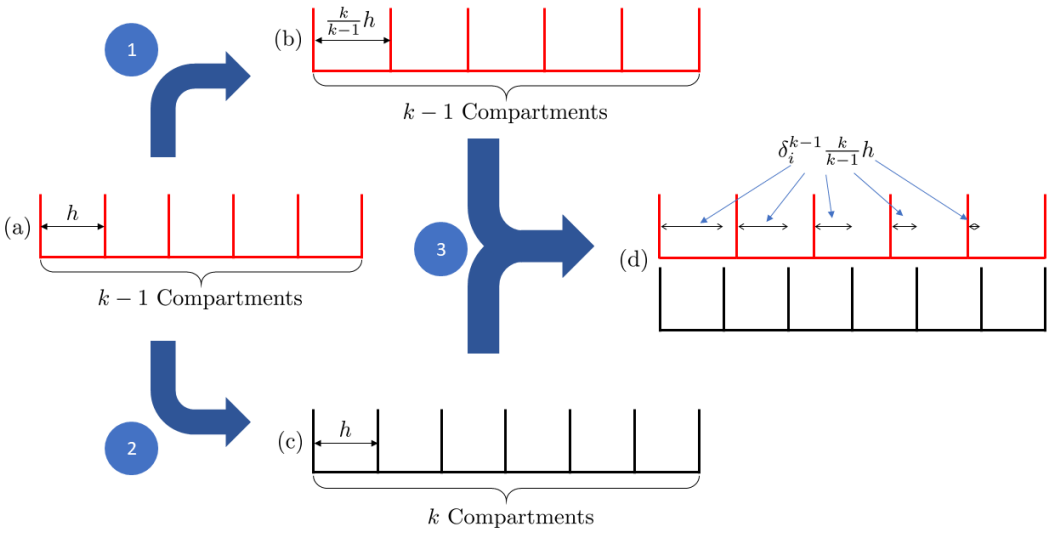


Figure 5. The process of domain growth for the stretching method. We start with $k-1$ compartments (a), and in step 1, stretch each of these compartments to be $k/(k-1)$ times their original length (b). This has the effect of increasing the domain length by h, a compartment's width. In step 2, we add a new compartment to the original $k-1$ compartments of size h (c), which again yields a domain that is a compartment's width bigger. In step 3, we redistribute the particles in the stretched $k-1$ compartments (c) into the k compartments of size h (b) by calculating the δ_i^{k-1} values, which tell us how much overlap there is between the two meshes (d). These δ_i^{k-1} values are treated as the probability for each particle in a stretched compartment to move to the renormalised compartment which overlaps its left-hand boundary. If a particle does not move to the renormalised compartment which overlaps with its left hand boundary then it moves to the renormalised compartment which overlaps with its right hand boundary.

$$\begin{aligned} &= \frac{k-i}{k-1}h. \\ \implies \delta_i^{k-1} &= \frac{k-i}{k}. \end{aligned} \tag{13}$$

4. Finally we calculate the new state $\boldsymbol{m}$ by drawing $k-1$ binomial random variables $b_i \sim \text{Bin}(r_i, \delta_i^{k-1})$ and calculating $\boldsymbol{m}$ as:

$$\boldsymbol{m} = \sum_{i=1}^{k-1} b_i \boldsymbol{e}_i + (r_i - b_i)\boldsymbol{e}_{i+1},$$

where the $\boldsymbol{e}_i$ are the standard k-dimensional basis vectors.

Algorithm 2: The stretching method

(2a) Initialise time $t = 0$ and set the final time $T > 0$. Initialise the number of compartments, k, and the particle numbers in each compartment $m_i,\ i \in \{1, ..., k\}$. Specify the size of compartment h, the jump rate $d = D/h^2$ and the growth rate ρ.

(2b) Calculate the propensity functions $\alpha_i^L = dm_i$, $\alpha_i^R = dm_i$ for left and right jumps from each compartment for $i = 1, ..., k$ with $\alpha_1^L = 0 = \alpha_k^R$, and set the propensity function for a growth event to be $\alpha^G = \rho k$. Calculate the sum of the propensity functions:

$$\alpha_0 = \sum_{i=1}^{k} \left(\alpha_i^L + \alpha_i^R\right) + \alpha^G.$$

(2c) Calculate the time until the next event by firstly drawing $u_1 \sim \text{Unif}(0, 1)$ and setting $\tau = 1/\alpha_0 \log(1/u_1)$. Set the time to be the next event that occurs $t \leftarrow t + \tau$.

(2d) Determine which event is next to occur by choosing one at random with probability proportional to the propensity function.

1. If the event corresponds to a left (resp. right) jump event from box i, remove a particle from compartment i and add one to compartment $i - 1$ (resp. $i + 1$).
2. If the event corresponds to a growth event: firstly, define the pregrowth state to be $\boldsymbol{r} \leftarrow \boldsymbol{m}$, calculate the overlap proportions $\delta_i^k = (k + 1 - i)/(k + 1)$, and use these in order to draw k binomial random variables $b_i \sim \text{Bin}(m_i, \delta_i^k)$. Create an extra compartment at the right end of the postgrowth domain (increasing k by 1 by setting $k \leftarrow k + 1$). Then set, for $j \in \{1, ..., k\}$:

$$m_j = \begin{cases} b_1, & \text{if } j = 1, \\ b_j + (r_{j-1} - b_{j-1}), & \text{if } j \in \{2, ..., k - 1\}, \\ r_{k-1} - b_{k-1}, & \text{if } j = k. \end{cases}$$

(2e) If $t < T$ then return to step **(2b)**, else end.

We assess the stretching method by initialising a uniform profile and test to see whether uniformity is maintained under the stretching domain growth method. In Figure 6, we can see that the stretching method performs very well in comparison to the original method.

3.1 Case study: Morphogen gradient

For our case study, we apply the original and stretching methods to the formation of a morphogen gradient on an exponentially growing domain [21, 22]. We begin with an initial domain of length L_0, which grows with rate ρ, and on which particles with density $u(x, t)$ move and interact. Particles diffuse with diffusion coefficient D, and they decay uniformly at a rate μ. There is also an influx of particles at the left-hand boundary, at rate $D\lambda$. We will only allow particles to jump between adjacent compartments and to interact within their own compartments. In order to compare the results of the stochastic simulation, we compute the solution to the associated mean-field PDE, which is given by:

$$\begin{aligned} \text{PDE}: \frac{\partial u}{\partial t} &= D\frac{\partial^2 u}{\partial x^2} - \frac{\partial(\rho x u)}{\partial x} - \mu u, && x \in (0, L_0 \exp(\rho t)),\ t > 0, \\ \text{BC}: \left.\frac{\partial u}{\partial x}\right|_{x=0} &= -\lambda,\ \left.\frac{\partial u}{\partial x}\right|_{x=L_0 \exp(\rho t)} = 0, && t > 0, \end{aligned} \tag{14}$$

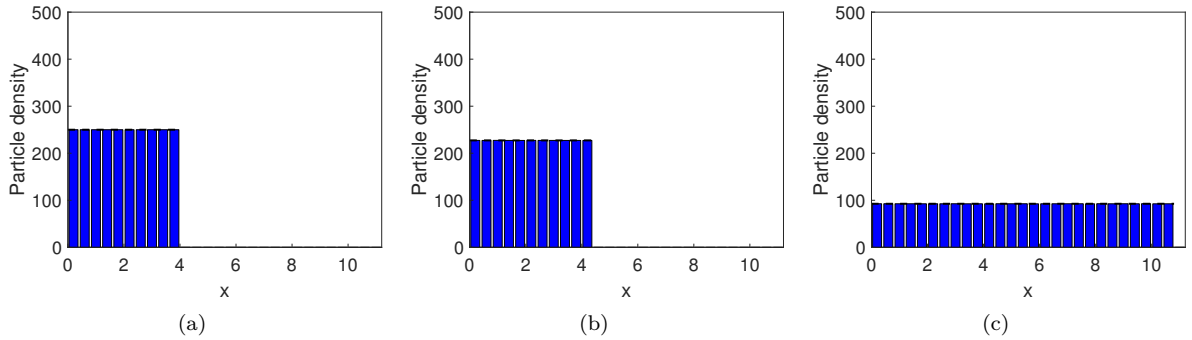


Figure 6. Three snapshots of the particle density on a domain growing according to the stretching method (a) initially, at time 0, (b) at time 10 and (c) at the final time 100. All descriptions and parameter values are the same as in Figure 2.

$$\text{IC} : u(x,0) = 0, \qquad x \in [0, L_0].$$

The PDE comprises three terms on the right-hand side — the first is the diffusive term, the second is dilution due to domain growth and the third is degradation of particles over the spatial domain [5]. The first boundary condition is the in-flux of particles at the left of the domain, while the second boundary condition is a reflective boundary. For our on-lattice simulations, diffusion is implemented by allowing particles to jump between neighbouring compartments with rate $d = D/h^2$. Particle degradation is achieved by allowing each particle in a compartment to be removed with rate μ, while production is included through a production reaction in the compartment adjacent to the left-hand boundary, as is described by Taylor et al. [7].

We simulate this system with exponential growth rate $\rho = 0.01$ on a domain of initial length $L_0 = 4$. We set the diffusion coefficient to be $D = 0.0025$, the influx rate is specified by setting $\lambda = 200$ and the degradation rate $\mu = 0.005$. We simulate until a final time, $t = 100$ and average over 50,000 repeats. The results are displayed in Figure 7.

As with the case of maintaining a uniform gradient, particle densities for the stretching method agree well with the associated mean-field PDE [15] (see Figures 7(d)-7(f)), however we still observe the same collection of particles at the boundaries for the original method (see Figures 7(a)-7(c)). This is particularly evident at the right-hand side of the domain, indicating that we are able to correctly simulate a reaction-diffusion system which incorporates first-order reactions using the stretching method. We anticipate that the extension to second- and higher-order reactions will yield similar results since the domain growth mechanisms is decoupled from the reaction mechanism.

3.2 Comparison of methods

In this section, we will investigate the two methods, and the parameter regimes in which the errors from the original method are acceptable due to the interplay between diffusion, the growth rate and initial domain length. We explore this in two ways. The first is through a heuristic argument.

The mean squared displacement describes how the variance in position of a Brownian particle changes in time. If multiple particles are initialised at the origin and diffuse for a time, t, then $\langle x^2 \rangle = 2Dt$, where the angled brackets denote an ensemble average of the squared distances from the origin. If a particle is to diffuse over the entire domain before the domain grows, then the squared distance from the origin would be $x^2 \sim L_0^2$. Likewise, the typical time frame for growth is given by setting $t \sim 1/\rho$. Substituting these into the expression for the mean squared displacement yields $D \sim L_0^2 \rho$. Therefore, we say we are

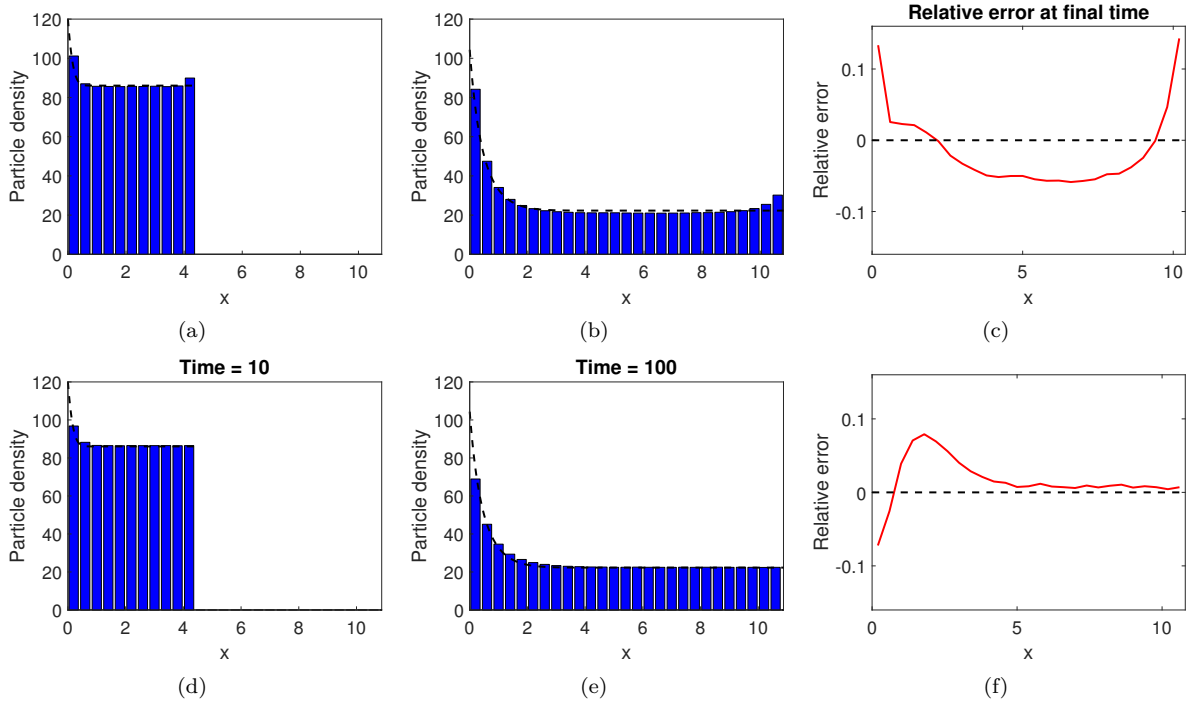


Figure 7. Two simulations of a morphogen gradient formation using (a)-(c) the original method and (d)-(f) the stretching method. In the first two columns, the blue bars represent the ensemble average of the stochastic algorithm over 50,000 repeats (a), (d) at time 10 and (b), (e) at time 100, while the black dashed line is the numerical solution to the PDE (14). In the final column we plot the relative errors between the on-lattice method and the PDE. All parameters are as in the text.

in a 'high diffusion' parameter regime when $D > L_0^2\rho$. When in the low diffusion regime ($D < L_0^2\rho$), particles are unable to spread and equilibrate before the domain grows, which is when we see the build up of particles in the original method.

We verify this heuristic result with a second argument. In order to do this, we have simulated a non-dimensionalised stochastic system and compared it to the solution of the mean-field continuum diffusion equation in order to determine a threshold value for diffusion. The results can be seen in Figure 8, where we plot the histogram distance error (HDE), which is given by the expression

$$\mathrm{HDE}(D^*) = \frac{1}{2}\sum_{n=1}^{N} |p_n^s(D^*) - p_n^m(D^*)|. \tag{15}$$

Here, $p_n^s(D^*)$ is the value of the normalised solution of the stochastic simulation (original or stretching method) at the final time with non-dimensional diffusion D^*, $p_n^m(D^*)$ is the solution for the associated non-dimensional mean-field PDE, and n indexes a common mesh on which we compare the two solutions. The non-dimensional diffusion parameter is equal to $D^* = D/(L_0^2\rho)$ where, as before, L_0 is the initial domain length and ρ is the exponential growth rate. From Figure 8(a), a value D^* greater than 1 yields similar HDE values for both the original and stretching methods. This indicates that, if $D > L_0^2\rho$ then the original method should have a similar performance to the stretching method. However, if the inequality is reversed, so that $D < L_0^2\rho$, then the stretching method should be used. We also note that the same

pattern is apparent when comparing the solutions of the mean equations (2) and (24), which can be seen in Figure 8(b). The HDE for the stretching method is exactly zero because it maintains uniformity precisely in the mean field.

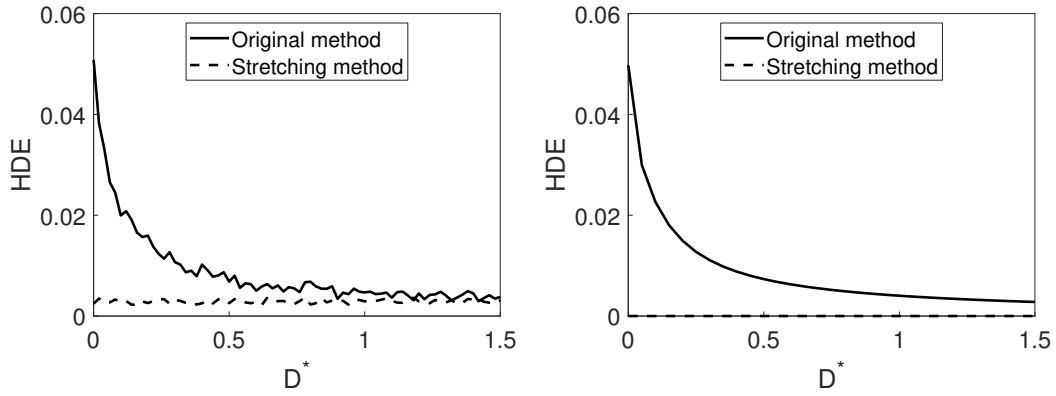


Figure 8. (a) HDEs for stochastic simulations of a non-dimensional diffusion mechanism on a growing domain, and their corresponding histogram distance errors (HDEs) for varying diffusion values. The solid line corresponds to the original method [15], while the dashed line denotes the stretching method. Each simulation is averaged over 5000 repeats. (b) A similar plot, but using the solutions of the mean equations (2) and (24). Note that the dotted line in this case is on top of the horizontal axis since there is no stochastic error in the mean equations and no bias in the stretching domain growth method.

4 Discussion

Domain growth mechanisms for on-lattice models are of importance for the accurate representation of many biological processes. We have demonstrated beyond doubt that the original method, suggested by Baker et al. [15], causes a build up of particles at the boundaries of the spatial domain (see Figure 2(c)). Consequently, we have developed a method for implementing domain growth when modelling reaction-diffusion systems at the mesoscale in order to correct this build-up. This technique involves stretching all compartments by a small amount (leading to the creation of a new compartment) and the appropriate re-distribution of the particles. We have demonstrated that this method agrees with the corresponding mean-field equations derived in the continuum limit, while maintaining a uniform profile, and have shown that it correctly models morphogen gradient formation on a growing domain.

The stretching method will be particularly useful when developing spatially extended hybrid methods on growing domains. These methods split the spatial domain into subdomains, in which different modelling paradigms are used, separated by an interface or overlap region [23]. We envisage that the stretching method can be used in the compartment-based subdomain of a hybrid model for reaction-diffusion on a growing domain, without causing a build-up of particles at the interface.

There are still several open questions regarding on-lattice domain growth whose answers go beyond the scope of this paper. The first of these concerns modelling domain growth in higher dimensions. To induce on-lattice domain growth in higher dimensions we can employ the following method by Binder et al. [24]. Consider, for the purposes of this example, a two-dimensional domain (although higher dimensional growth is straightforward to implemented by analogy). In order to maintain a rectangular domain, a "growth event" must increase the total number of rows or columns by one. For example, when a growth event is chosen to occur in the vertical direction, we must increase in the number of rows. To do this,

we temporarily treat each column as its own one-dimensional domain, and implement a single vertical growth event in each column, independent of the others. Doing this for every column results in the whole domain increasing in height by a single row.

We have simulated such a domain growth process using the original method of Baker et al. [15] to implement the independent row or column elongations when carrying out a horizontal or vertical (respectively) growth event. Specifically, for clarity, we carry out horizontal and vertical growth events simultaneously, which maintains the aspect ratio of the initially square domain we begin with. Diffusion of particles is turned off in order to clearly demonstrate the bias induced by this two-dimensional version of domain growth as illustrated in Figure 9. The same effect that we have observed in one dimension (namely a preponderance of particles towards the boundaries of the domain) is also present in higher dimensions. Extending the stretching method will provide a straightforward fix to this problem in higher dimensions. Overlap fractions will be calculated as ratios of (hyper-)volumes as opposed to ratios of lengths, and particles in a pregrowth compartment will be distributed between multiple overlapping postgrowth compartments using multinomial distributions (the natural generalisation of the binomial distributions used in the one-dimensional case).

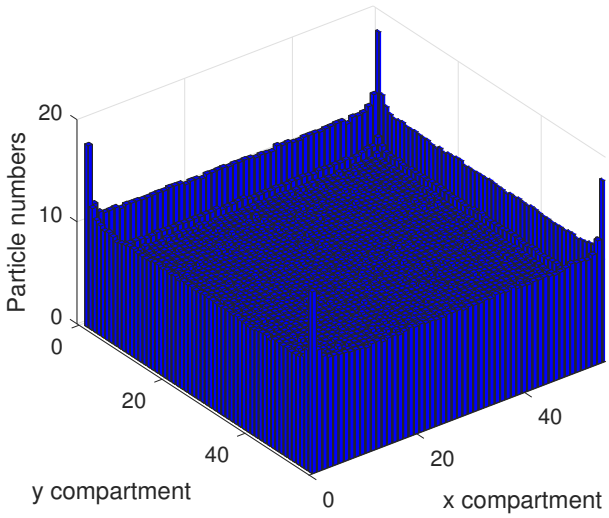


Figure 9. A simulation of the domain growth method of Baker et al. [15] extended to two dimensions. The method of domain growth is described in the text. The domain contains 1,000 particles, and particle redistribution is due to domain growth events alone as diffusion is set to zero. The growth rate is $\rho = 0.01$, and the particle densities are averaged over 10,000 independent repeats.

Whilst domain growth is important, equally, domain shrinkage is also of significant biological interest. Domain shrinkage through directed apoptosis (programmed cell death) is an important component of many wound healing processes [25, 26] for example. Without further investigation it is not immediately clear whether domain shrinkage implemented by the removal of a randomly chosen compartment (in analogy to the domain *growth* method of Baker et al. [15] and as implemented by Yates [3]) would induce bias in cell densities. In contrast we are confident that implementing domain shrinkage into the stretching method by considering pre- and post-shrink overlap regions compartments will not induce bias. Nevertheless these hypotheses remain to be tested.

Finally, it is of interest to adapt the stretching method to account for non-uniform growth, which has been shown to be important in biological scenarios [27]. All of the examples we have presented have implemented uniform growth — in which all regions of space grow at the same rate. The stretching method could be adapted to incorporate non-uniform growth by splitting the domain into groups of

compartments each of which have a different growth rate. The stretching method can then be used on each of the groups individually.

Many authors have used the method presented by Baker et al. [15] to incorporate growth into on-lattice simulations of reaction-diffusion processes. For example, Woolley et al. [18] investigate the role that domain growth plays on modelling stochastic reaction-diffusion systems. Thompson et al. [19] explore cell migration and adhesion during biological development, while tissue growth and shrinkage are studied by Yates [3]. However, as demonstrated in this paper, particularly in the case of low diffusivity, the inherent bias in the domain growth method suggests that the conclusions drawn from these studies may require re-evaluation. We suggest that the 'stretching' domain growth method we propose in this paper is an appropriate alternative with which to re-evaluate these results and which should be employed in future studies of reaction-diffusion processes on growing domains.

Acknowledgements

CAS is supported by a scholarship from the EPSRC Centre for Doctoral Training in Statistical Applied Mathematics at Bath (SAMBa), under the project EP/L015684/1. CM is grateful to EPSRC for support through the fellowship EP/R022186/1.

Appendices

A Deriving equation (5)

Within this appendix, we derive the expression in equation (5):

$$\prod_{n=a}^{b}\left(1+\frac{c}{n}\right)\approx\frac{\Gamma(a)}{\Gamma(a+c)e^{c}}b^{c}, \tag{5}$$

which holds for $a, b \in \mathbb{N}$ such that $a < b$ and b is large, and for $c \in \mathbb{R}$. We start by representing the product on the left-hand side of equation (5) using gamma functions:

$$\begin{aligned}\prod_{n=a}^{b}\left(1+\frac{c}{n}\right) &= \prod_{n=a}^{b}\left(\frac{n+c}{n}\right)\\ &= \frac{\Gamma(a)\Gamma(b+c+1)}{\Gamma(a+c)\Gamma(b+1)}.\end{aligned} \tag{A1}$$

We apply Stirling's approximation, which says that for large z, $\Gamma(z) \approx \sqrt{2\pi(z-1)}(z-1)^{z-1}e^{-(z-1)}$. Applying this to equation (A1), we obtain:

$$\begin{aligned}\prod_{n=a}^{b}\left(1+\frac{c}{n}\right) &\approx \frac{\Gamma(a)}{\Gamma(a+c)}\frac{\sqrt{2\pi(b+c)}}{\sqrt{2\pi b}}\frac{(b+c)^{b+c}}{b^{b}}\frac{e^{b}}{e^{b+c}}\\ &= \frac{\Gamma(a)}{\Gamma(a+c)}\sqrt{\frac{b+c}{b}}\left(\frac{b+c}{b}\right)^{b}(b+c)^{c}e^{-c}.\end{aligned}$$

We assume that b is large, and thus, $b + c \approx b$. Applying this:

$$\prod_{n=a}^{b}\left(1+\frac{c}{n}\right)\approx\frac{\Gamma(a)}{\Gamma(a+c)}\sqrt{\frac{b}{b}}\left(\frac{b}{b}\right)^{b}b^{c}e^{-c}$$

$$= \frac{\Gamma(a)}{\Gamma(a+c)e^c} b^c.$$

B Deriving equation (9)

Within this appendix, we derive the expression in equation (9):

$$\sum_{n=2}^{k-1} \left(\prod_{j=n+1}^{k} \left(1 - \frac{3/2}{j}\right) \right) M_1^n \left(\frac{1/2}{n}\right) \approx \frac{1}{2} \frac{1}{\sqrt{\pi} e^{-1/2}} k^{-1/2} - \frac{1}{\sqrt{\pi} e^{-1/2}} k^{-3/2}, \tag{9}$$

where the approximation is for large values of k.

$$\begin{aligned}
\sum_{n=2}^{k-1} \left(\prod_{j=n+1}^{k} \left(1 - \frac{3/2}{j}\right) \right) M_1^n \left(\frac{1/2}{n}\right) &= \sum_{n=2}^{k-1} \left(\prod_{j=1}^{k} \left(1 - \frac{3/2}{j}\right) \right) \left(\prod_{j=1}^{n} \left(1 - \frac{3/2}{j}\right) \right)^{-1} M_1^n \left(\frac{1/2}{n}\right) \\
&= \left(\prod_{j=1}^{k} \left(1 - \frac{3/2}{j}\right) \right) \sum_{n=2}^{k-1} \left(\prod_{j=1}^{n} \left(1 - \frac{3/2}{j}\right) \right)^{-1} M_1^n \left(\frac{1/2}{n}\right) \\
&\approx k^{-3/2} \sum_{n=2}^{k-1} \left(n^{-3/2}\right)^{-1} M_1^n \left(\frac{1/2}{n}\right),
\end{aligned}$$

where, in the final approximation, we have used equation (5) in order to simplify the two products. Simplifying yields:

$$\sum_{n=2}^{k-1} \left(\prod_{j=n+1}^{k} \left(1 - \frac{3/2}{j}\right) \right) M_1^n \left(\frac{1/2}{n}\right) \approx k^{-3/2} \sum_{n=2}^{k-1} n^{3/2} M_1^n \left(\frac{1/2}{n}\right).$$

We now utilise equation (6) (approximation is better for large k), in order to obtain:

$$\begin{aligned}
\sum_{n=2}^{k-1} \left(\prod_{j=n+1}^{k} \left(1 - \frac{3/2}{j}\right) \right) M_1^n \left(\frac{1/2}{n}\right) &\approx k^{-3/2} \sum_{n=2}^{k-1} n^{3/2} \frac{1}{\sqrt{\pi} e^{-1/2}} n^{-1/2} \frac{1}{2} n^{-1} \\
&= k^{-3/2} \frac{1}{\sqrt{\pi} e^{-1/2}} \sum_{n=2}^{k-1} \frac{1}{2} \\
&= k^{-3/2} \frac{1}{\sqrt{\pi} e^{-1/2}} \frac{1}{2} (k-2) \\
&= \frac{1}{2} \frac{1}{\sqrt{\pi} e^{-1/2}} k^{-1/2} - \frac{1}{\sqrt{\pi} e^{-1/2}} k^{-3/2}.
\end{aligned}$$

C Discussion on local methods

Throughout Section 2, we have demonstrated that at low diffusion levels, the method presented by Baker et al. [15] fails to correctly model the growth of a uniform profile. This is a local method, which means that on every occasion that the domain is due to grow, we choose a compartment at random at which

the growth event occurs. Over many repeats of the same process, different compartments will be chosen and so, when averaging over these repeats, each compartment is chosen an equal number of times (when considering only a single growth event).

There are three elements that define a local method:

1. The probability of choosing each compartment,
2. The direction of growth,
3. The redistribution of particles.

In the case of Baker et al. [15], we (1) choose each compartment uniformly at random, (2) always grow to the right and (3) redistribute the particles using a symmetric binomial distribution. We now discuss which of these we can change in order to create a local method that correctly maintains a uniform profile on a growing domain.

The simplest way of creating a different local method is to change one of the three elements. However, it can be shown that changing only a single element does not yield the expected uniform growth. As a result, the next simplest is to change two of the elements, whilst fixing one. One such way would be to (1) set the probability of choosing each compartment to be general, (2) set the direction of growth to be left or right with a probability of a half each, and (3) redistribute the particles using a binomial distribution with a generic probability of success. These properties yield an algorithm which has k degrees of freedom (where there are $k-1$ pregrowth compartments), which can be used in order to solve a series of equations to ensure that a uniform profile is maintained.

In order to calculate the values for the probability distribution, and the probability for success in the binomial distribution when redistributing particles post-split, we can write a series of equations that relate the pre- and postgrowth states, on average. Using these, we are able to use numerical optimisation techniques in order to find the optimum values for the unknown parameters. However, while this method fixes the main issue with the original method, it introduces some new ones. Firstly, the centre of growth is no longer fixed at one end of the domain. This may cause problems for certain methodological applications (e.g. employing the method for spatial hybrid methods [28, 29, 30]) or for some biological applications in which tissues genuinely grow from a fixed origin at one end of the domain grow from one end of the domain (e.g. hyphal tip growth [31]). Secondly, we have to solve an overdetermined system for every possible number of compartments that might occur. This can be computationally expensive, especially for large compartment numbers. In practice, the unknown parameters can be computed and stored *a priori*, although, if the timing of domain growth events is stochastic, it may not be clear in advance exactly how many compartments the domain will comprise.

We also note that we have changed the direction of growth from being always to the right in the original method, to being left or right with equal probability. This is because there is no probability distribution for the compartments together with redistribution probability that maintains a uniform profile when the direction of growth is always the same.

D Justifying the stretching method

In Section 3, we introduced the stretching method as an alternative to the original method presented by Baker et al. [15]. In order to demonstrate that the original method fails, we have simulated the corresponding mean equations, and also analytically calculated the density of particles in each compartment with a large number of compartments. In this section, we will conduct a similar analysis of the stretching method.

D.1 The mean equations

In this section, we will calculate the mean equations for the stretching method by firstly considering the master equation for the process. We will then simulate the solutions to the mean equations, demonstrating that the solution remains uniform.

We begin by defining the sets $\mathcal{M}_k^N = \left\{\boldsymbol{m} \in \mathbb{N}^k : \sum_{i=1}^k m_i = N\right\}$. This is the set of all state vectors when there are k compartments and N particles in the system in total. We would like an expression for the probability that there are k compartments in total, and the state variable is $\boldsymbol{m} \in \mathcal{M}_k^N$ with N a fixed integer. We call this probability $p(\boldsymbol{m}, k, t)$ as we did in Section 2. Then, defining $\pi(\boldsymbol{m}|\boldsymbol{r})$ to be the transition probability from state $\boldsymbol{r}$ to state $\boldsymbol{m}$:

$$\frac{\mathrm{d}p}{\mathrm{d}t}(\boldsymbol{m}, k, t) = \rho(k-1) \sum_{\boldsymbol{r} \in \mathcal{M}_{k-1}^N} [p(\boldsymbol{r}, k-1, t)\pi(\boldsymbol{m}|\boldsymbol{r})] - \rho k p(\boldsymbol{m}, k, t). \tag{16}$$

Here, the summand in the first term of the right hand side represents the rate of moving from a state $\boldsymbol{r} \in \mathcal{M}_{k-1}^N$ to the state $\boldsymbol{m} \in \mathcal{M}_k^N$. The second term is the rate at which the process leaves the state $\boldsymbol{m}$.

We next derive the mean equations. Similarly to the original method, define:

$$M_i^k(t) = \sum_{\boldsymbol{m} \in \mathcal{M}_k^N} m_i p(\boldsymbol{m}, k, t). \tag{17}$$

Multiplying the CME (16) by m_i, summing over all possible $\boldsymbol{m} \in \mathcal{M}_k^N$, and applying equation (17), we obtain:

$$\frac{\mathrm{d}M_i^k}{\mathrm{d}t}(t) = \rho(k-1) \sum_{\boldsymbol{m} \in \mathcal{M}_k^N} \sum_{\boldsymbol{r} \in \mathcal{M}_{k-1}^N} [m_i p(\boldsymbol{r}, k-1, t)\pi(\boldsymbol{m}|\boldsymbol{r})] - \rho k M_i^k(t). \tag{18}$$

From now on, we will drop the range of the sums, and simply write $\boldsymbol{m}$ or $\boldsymbol{r}$ and implicitly assume we are summing over the correct sets in order to simplify the notation. Recall that for each $\boldsymbol{r}$, there is an associated vector of binomial random variables $\boldsymbol{b}$ that re-distributes the particles from the pregrowth state to the postgrowth state. Further, for any $j \in \{1, ..., k-1\}$, the probability of drawing b_j is given by the probability mass function:

$$\mathbb{P}(b_j|\boldsymbol{r}) = \binom{r_j}{b_j} \left(\delta_j^{k-1}\right)^{b_j} (1 - \delta_j^{k-1})^{r_j - b_j}, \tag{19}$$

where δ_j^{k-1} is the 'overlap' proportion defined in equation (13). We can then use the relationships set out in Algorithm 2, namely:

$$m_j = \begin{cases} b_1, & \text{if } j = 1, \\ b_j + (r_{j-1} - b_{j-1}), & \text{if } j \in \{2, ..., k-1\}, \\ r_{k-1} - b_{k-1}, & \text{if } j = k. \end{cases} \tag{20}$$

Assuming that $i \in \{2, ..., k-1\}$ (a similar argument can be applied to the case $i \in \{1, k\}$) and substituting the relationships in (20) into the sum in equation (18), we obtain:

$$\begin{aligned} \sum_{\boldsymbol{m}} \sum_{\boldsymbol{r}} m_i p(\boldsymbol{r}, k-1, t)\pi(\boldsymbol{m}|\boldsymbol{r}) &= \sum_{\boldsymbol{m}} \sum_{\boldsymbol{r}} (r_{i-1} - b_{i-1} + b_i) p(\boldsymbol{r}, k-1, t)\pi(\boldsymbol{m}|\boldsymbol{r}) \\ &= \sum_{\boldsymbol{r}} \sum_{\boldsymbol{m}} r_{i-1} p(\boldsymbol{r}, k-1, t)\pi(\boldsymbol{m}|\boldsymbol{r}) - \sum_{\boldsymbol{r}} \sum_{\boldsymbol{m}} (b_{i-1} - b_i) p(\boldsymbol{r}, k-1, t)\pi(\boldsymbol{m}|\boldsymbol{r}), \end{aligned} \tag{21}$$

where in the final step, we have split the sum and also switched the order of summation. We will now make use of the binomial random variables (the b_j's). We note that in order to transition from the

pregrowth state $\boldsymbol{r} \in \mathcal{M}_{k-1}^N$ to the postgrowth state $\boldsymbol{m} \in \mathcal{M}_k^N$, we need to find a vector $\boldsymbol{b}$. However, only a small subset of these $\boldsymbol{b}$ vectors have a non-zero probability of occurring. Therefore, the sum over $\boldsymbol{m}$ can be re-written as a sum over the possible $\boldsymbol{b}$ values that have a non-zero probability. Using this, and substituting for $\pi(\boldsymbol{m}|\boldsymbol{r})$ the specific probability $\mathbb{P}(\boldsymbol{b}|\boldsymbol{r})$ of the binomial redistribution which takes us from state $\boldsymbol{r}$ to state $\boldsymbol{m}$, equation (21) becomes:

$$\begin{aligned}\sum_{\boldsymbol{r}}\sum_{\boldsymbol{m}} m_i p(\boldsymbol{r},k-1,t)\pi\left(\boldsymbol{m}|\boldsymbol{r}\right) &= \sum_{\boldsymbol{r}}\sum_{\boldsymbol{m}} r_{i-1} p(\boldsymbol{r},k-1,t)\pi\left(\boldsymbol{m}|\boldsymbol{r}\right) - \sum_{\boldsymbol{r}}\sum_{\boldsymbol{m}} (b_{i-1}-b_i) p(\boldsymbol{r},k-1,t)\pi\left(\boldsymbol{m}|\boldsymbol{r}\right) \\ &= \sum_{\boldsymbol{r}}\sum_{\boldsymbol{b}} r_{i-1} p(\boldsymbol{r},k-1,t)\mathbb{P}\left(\boldsymbol{b}|\boldsymbol{r}\right) - \sum_{\boldsymbol{r}}\sum_{\boldsymbol{b}} (b_{i-1}-b_i) p(\boldsymbol{r},k-1,t)\mathbb{P}\left(\boldsymbol{b}|\boldsymbol{r}\right).\end{aligned} \tag{22}$$

Here $\mathbb{P}(\boldsymbol{b}|\boldsymbol{r}) = \prod_{j=1}^{k-1}\mathbb{P}(b_j|\boldsymbol{r})$ (where $\mathbb{P}(b_j|r_j)$ are given in equation (19)), and the sum over $\boldsymbol{b}$ is a sum over the set $\{\boldsymbol{b} : b_j \in \{0,...,r_j\}\ \forall j \in \{1,...,k-1\}\}$. We now re-arrange the two sums in expression (22):

$$\begin{aligned}&\sum_{\boldsymbol{r}}\sum_{\boldsymbol{b}} r_{i-1} p(\boldsymbol{r},k-1,t)\mathbb{P}\left(\boldsymbol{b}|\boldsymbol{r}\right) - \sum_{\boldsymbol{r}}\sum_{\boldsymbol{b}} (b_{i-1}-b_i) p(\boldsymbol{r},k-1,t)\mathbb{P}\left(\boldsymbol{b}|\boldsymbol{r}\right) \\ &\quad= \sum_{\boldsymbol{r}} r_{i-1} p(\boldsymbol{r},k-1,t)\left[\sum_{\boldsymbol{b}}\mathbb{P}(\boldsymbol{b}|\boldsymbol{r})\right] - \sum_{\boldsymbol{r}} p(\boldsymbol{r},k-1,t)\left[\sum_{\boldsymbol{b}}(b_{i-1}-b_i)\mathbb{P}(\boldsymbol{b}|\boldsymbol{r})\right].\end{aligned}$$

The sum in the square brackets in the first term is equal to 1 as it is the sum of a probability distribution and the sum in the square brackets in the second term is a difference of two expectations:

$$\begin{aligned}&\sum_{\boldsymbol{r}} r_{i-1} p(\boldsymbol{r},k-1,t)\left[\sum_{\boldsymbol{b}}\mathbb{P}(\boldsymbol{b}|\boldsymbol{r})\right] - \sum_{\boldsymbol{r}} p(\boldsymbol{r},k-1,t)\left[\sum_{\boldsymbol{b}}(b_{i-1}-b_i)\mathbb{P}(\boldsymbol{b}|\boldsymbol{r})\right] \\ &\quad= \sum_{\boldsymbol{r}} r_{i-1} p(\boldsymbol{r},k-1,t) - \sum_{\boldsymbol{r}} p(\boldsymbol{r},k-1,t)\left[\mathbb{E}[b_{i-1}|\boldsymbol{r}] - \mathbb{E}[b_i|\boldsymbol{r}]\right].\end{aligned}$$

Finally, using the standard expectation for the binomial distribution, and bringing terms together, we find that:

$$\begin{aligned}\sum_{\boldsymbol{m}}\sum_{\boldsymbol{r}} m_i p(\boldsymbol{r},k-1,t)\pi\left(\boldsymbol{m}|\boldsymbol{r}\right) &= (1-\delta_{i-1}^{k-1})\sum_{\boldsymbol{r}} r_{i-1} p(\boldsymbol{r},k-1,t) + \delta_i^{k-1}\sum_{\boldsymbol{r}} r_i p(\boldsymbol{r},k-1,t) \\ &= (1-\delta_{i-1}^{k-1})M_{i-1}^{k-1} + \delta_i^{k-1}M_i^{k-1},\end{aligned} \tag{23}$$

where the final equality uses the definition (17). Substituting this expression into equation (18) yields the mean-field density evolution equations for the stretching method:

$$\frac{\mathrm{d}M_i^k}{\mathrm{d}t} = \rho(k-1)\left[(1-\delta_{i-1}^{k-1})M_{i-1}^{k-1} + \delta_i^{k-1}M_i^{k-1}\right] - \rho k M_i^k. \tag{24}$$

D.2 Analytical density

Here we will use a similar approach to Section 3 in order to calculate the average density of particles when the domain grows according to the stretching method. We let u_i^k be the random density of particles in compartment i when there are k compartments in total, $M_i^k = \mathbb{E}[u_i^k]$, and $q(u_i^k, \delta_i^k)$ a realisation from a general probability distribution with mean value $M_i^k\delta_i^k$. The $q(u_i^k, \delta_i^k)$ values denote the density that is redistributed into postgrowth compartment i from pregrowth compartment i, analogous to the $\mathbb{P}(b_i|r_i)$ in equation (19) for the discretised process. Then the u_i^k's are related by the following recursive relation:

$$u_i^k = q\left(u_i^{k-1}, \delta_i^{k-1}\right) + \left(u_{i-1}^{k-1} - q\left(u_{i-1}^{k-1}, \delta_{i-1}^{k-1}\right)\right) \tag{25}$$

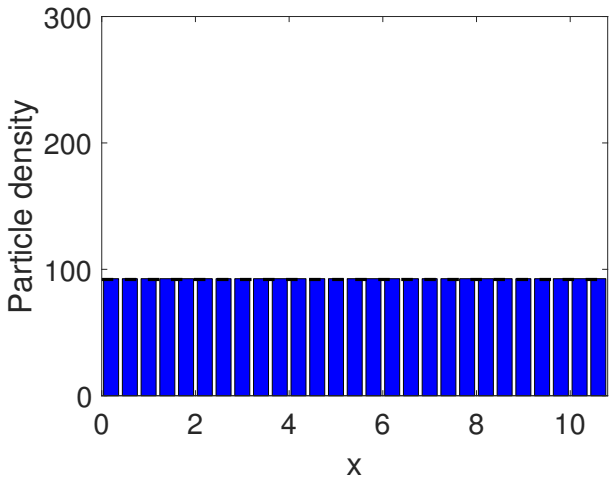


Figure 10. The solution to the mean-field equations (24). All parameters are the same as Figure 3(b).

with an initial condition $u_1^1 = 1$. Here, the first term on the right-hand side is the fraction of the density that is provided to compartment i on the postgrowth domain by the pregrowth compartment to the right, while the second term is the density provided from the left pregrowth compartment. We now take expectations to yield the recursive equations:

$$M_i^k = \delta_i^{k-1} M_i^{k-1} + (1 - \delta_{i-1}^{k-1}) M_{i-1}^{k-1}. \tag{26}$$

We show that, under this recursion relation, $M_i^k = 1/k$ by induction on the number of compartments, k. Clearly, using the initial condition, we have the base case $M_1^1 = 1$. Now assume that $M_i^{k-1} = 1/(k-1)$ for all $i \in \{1, ..., k-1\}$. Then, using (26):

$$\begin{aligned} M_i^k &= \delta_i^{k-1} M_i^{k-1} + (1 - \delta_{i-1}^{k-1}) M_{i-1}^{k-1} \\ &= \frac{\delta_i^{k-1}}{k-1} + \frac{1 - \delta_{i-1}^{k-1}}{k-1} \\ &= \frac{k - i + (k - (k - i - 1))}{k(k-1)} \\ &= \frac{1}{k}. \end{aligned}$$

Here, the second line uses the inductive hypothesis and the third line employs the definition of δ_i^{k-1}. This completes the inductive step, and hence the proof that $M_i^k = 1/k$.

References

1. R. McLennan, L. Dyson, K.W. Prather, J.A. Morrison, R.E. Baker, P.K. Maini, and Kulesa P.M. Multiscale mechanisms of cell migration during development: theory and experiment. *Development*, 139(16):2935–2944, 2012.

2. P.M. Kulesa, C.M. Bailey, J.C. Kasemeier-Kulesa, and R. McLennan. Cranial neural crest migration: new rules for an old road. *Dev. Biol.*, 344(2):543–554, 2010.

3. C.A. Yates. Discrete and continuous models for tissue growth and shrinkage. *J. Theor. Biol.*, 350: 37–48, 2014.

4. L. Wolpert, C. Tickle, and A.M. Arias. *Principles of development*. Oxford University Press, 5 edition, 2015.

5. E.J. Crampin, E.A. Gaffney, and P.K. Maini. Reaction and diffusion on growing domains: scenarios for robust pattern formation. *Bull. Math. Biol.*, 61(6):1093–1120, 1999.

6. E.J. Crampin. *Reaction-diffusion patterns on growing domains*. PhD thesis, University of Oxford, 2000.

7. P.R. Taylor, R.E. Baker, and C.A. Yates. Deriving appropriate boundary conditions, and accelerating position-jump simulations, of diffusion using non-local jumping. *Phys. Biol.*, 12(1):016006, 2015.

8. S.A. Isaacson. A convergent reaction-diffusion master equation. *J. Chem. Phys.*, 139(5):054101, 2013.

9. D.T. Gillespie. Exact stochastic simulation of coupled chemical reactions. *J. Phys. Chem.*, 81(25): 2340–2361, 1977.

10. M.A. Gibson and J. Bruck. Efficient exact stochastic simulation of chemical systems with many species and many channels. *J. Phys. Chem. A.*, 104(9):1876–1889, 2000.

11. J. Elf and M. Ehrenberg. Spontaneous separation of bi-stable biochemical systems into spatial domains of opposite phases. *Syst. Biol.*, 1(2):230–236, 2004.

12. J. Hattne, D. Fange, and J. Elf. Stochastic reaction-diffusion simulation with MesoRD. *Bioinformatics*, 21(12):2923–2924, 2005.

13. J.M. McCollum, G.D. Peterson, C.D. Cox, M.L. Simpson, and N.F. Samatova. The sorting direct method for stochastic simulation of biochemical systems with varying reaction execution behavior. *Comput. Biol. Chem.*, 30(1):39–49, 2006.

14. A.M. Smith, R.E. Baker, D. Kay, and P.K. Maini. Incorporating chemical signalling factors into cell-based models of growing epithelial tissues. *J. Math. Biol.*, 65(3):441–463, 2012.

15. R.E. Baker, C.A. Yates, and R. Erban. From microscopic to macroscopic descriptions of cell migration on growing domains. *Bull. Math. Biol.*, 72(3):719–762, 2010.

16. C.A. Yates and R.E. Baker. Importance of the voronoi domain partition for position-jump reaction-diffusion processes on non-uniform rectilinear lattices. *Phys. Rev. E*, 88(5):054701, 2013.

17. See Supplemental Material at [URL will be inserted by publisher] for the accompanying MATLAB code.

18. T.E. Woolley, R.E. Baker, E.A. Gaffney, and P.K. Maini. Stochastic reaction and diffusion on growing domains: understanding the breakdown of robust pattern formation. *Phys. Rev. E*, 84(4): 046216, 2011.

19. R.N. Thompson, C.A. Yates, and R.E. Baker. Modelling cell migration and adhesion during development. *Bull. Math. Biol.*, 74(12):2793–2809, 2012.

20. C.A. Yates, R.E. Baker, R. Erban, and P.K. Maini. Going from microscopic to macroscopic on non-uniform growing domains. *Phys. Rev. E*, 86(2):021921, 2012.

21. L. Wolpert. Positional information and the spatial pattern of cellular differentiation. *J. Theor. Biol.*, 25(1):1–47, 1969.

22. K.J. Painter, P.K. Maini, and H.G. Othmer. Stripe formation in juvenile pomacanthus explained by a generalized turing mechanism with chemotaxis. *Proc. Natl. Acad. Sci. USA*, 96(10):5549–5554, 1999.

23. C.A. Smith and C.A. Yates. Spatially extended hybrid methods: a review. *J. Roy. Soc. Interface*, 15(139), 2018.

24. B.J. Binder, K.A. Landman, M.J. Simpson, M. Mariani, and D.F. Newgreen. Modeling proliferative tissue growth: a general approach and an avian case study. *Phys. Rev. E*, 78(3):031912, 2008.

25. D.G. Greenhalgh. The role of apoptosis in wound healing. *Int. J. Biochem. Cell Biol.*, 30(9): 1019–1030, 1998.

26. F. Grinnell, M. Zhu, M.A. Carlson, and J.M. Abrams. Release of mechanical tension triggers apoptosis of human fibroblasts in a model of regressing granulation tissue. *Exp. Cell. Res.*, 248(2): 608–619, 1999.

27. M.J. Simpson, K.A. Landman, and D.F. Newgreen. Chemotactic and diffusive migration on a nonuniformly growing domain: numerical algorithm development and applications. *J. Comput. Appl. Math.*, 192(2):282–300, 2006.

28. C.A. Yates and M.B. Flegg. The pseudo-compartment method for coupling partial differential equation and compartment-based models of diffusion. *J. Roy. Soc. Interface*, 12(106):20150141, 2015.

29. M.B. Flegg, S. Hellander, and R. Erban. Convergence of methods for coupling of microscopic and mesoscopic reaction-diffusion simulations. *J. Comput. Phys.*, 289(C):1–17, May 2015. ISSN 0021-9991.

30. C.A. Smith and C.A. Yates. The auxiliary region method: A hybrid method for coupling a PDE to Brownian-based dynamics for reaction-diffusion systems. *R. Soc. Open Sci.*, 5(8), 2018.

31. A. Goriely and M. Tabor. Mathematical modeling of hyphal tip growth. *Fungal Biol. Rev.*, 22(2): 77–83, 2008.

4.2 Conclusions

In this paper, we have created an alternative algorithm to that of Baker et al. (2010) for modelling reaction-diffusion processes at the mesoscale on a growing domain. We have demonstrated that our method is both accurate for modelling reaction-diffusion processes, and that it does not suffer from problems when particles diffuse at a much slower rate than they are spread out by the domain growth. This is something that the original method is unable to capture, and instead causes a build up of particles at each of the boundaries rather than generating a uniform distribution.

As a result of this expected behaviour, this method provides a valuable tool for anyone wishing to simulate reaction-diffusion processes using the mesoscopic approach. In particular, it will be of great importance for modellers wishing to represent processes which are characterised by a low-diffusion regime.

Having established an equivalence between the stretching method and the standard diffusion equation on a uniformly growing domain, the next natural question for us to ask in the context of this thesis is how does this new method couple to its corresponding PDE as part of a hybrid method. Applying the stretching method to a spatially extended hybrid method has the potential to create algorithms that are both accurate and efficient, so that they can be applied to more realistic scenarios.

The next chapter comprises another paper that establishes equivalence frameworks, which together with the paper presented in this chapter, will be heavily employed in Chapter 6 in order to create hybrid methods on growing domains. Whilst our work in this chapter has focussed on one of the three modelling paradigms, the paper in the next chapter concerns boundary conditions. In particular, we are interested in the question of how we simulate Robin boundary conditions, defined at the macroscale, at both the mesoscale and microscale on growing domains.

Chapter 5

Robin boundary conditions on growing domains: equating PDE to stochastic representations

Within this chapter, we develop an equivalence framework for general Robin boundary conditions on uniformly growing domains. We define the boundary condition for the PDE, and derive appropriate rules for both the compartment-based (mesoscopic) and individual-based (microscopic) approaches in order to match the PDE boundary condition. The approach used here is adapted from those of Singer et al. (2008), Erban and Chapman (2007), and Taylor et al. (2015) extended to deal with domain growth and an influx condition for the microscale. We demonstrate, through a series of test problems, that the conditions we derived for equivalence give particle density profiles that match those of the associated PDE. These parameters describe the rates at which particles should be added to the domain at the boundary, and probabilities that, given that a particle has hit the boundary, that it is absorbed (or that a new particle is added).

In the context of the thesis, this paper provides the equivalence framework for one of the main case studies for the paper in Chapter 6. This example is used in order to demonstrate that the three growing hybrid methods are able to effectively represent situations in which the gradient over the interface is non-zero.

5.1 Outline of paper

In Section 1, we introduce the problem, describing the PDE and the associated continuum boundary condition that we aim to replicate with the stochastic representations. Section 2 contains a calculation that links mesoscopic probabilities and rates required in order to match this representation to the PDE, while Section 3 contains a long proof which does the same calculation, but for the individual-based paradigm. Section 4 contains four examples which demonstrate that the derived equivalences are appropriate. Finally, in Section 5 we end with some concluding remarks.

Appendix 6B: Statement of Authorship

This declaration concerns the article entitled:			
Robin boundary conditions on growing domains: equating PDE to stochastic representations			
Publication status (tick one) **Draft manuscript** [•] **Submitted** [ ] **In review** [ ] **Accepted** [ ] **Published** [ ]			
Publication details (reference)	Authors – Cameron A. Smith, Christian A. Yates		
Copyright status (tick the appropriate statement) I hold the copyright for this material [•] Copyright is retained by the publisher, but I have been given permission to replicate the material here [ ]			
Candidate's contribution to the paper (provide details, and also indicate as a percentage)	All calculations have been performed by the author of this thesis (100%). All numerical computations and simulations have been completed by the author of this thesis (100%). All authors contributed equally to the presentation of the content (50%).		
Statement from Candidate	This paper reports on original research I conducted during the period of my Higher Degree by Research candidature.		
Signed		**Date**	04/06/2021

Robin boundary conditions on growing domains: equating PDE to stochastic representations

Cameron A. Smith[1,*], Christian A. Yates[1]
[1]**Centre for Mathematical Biology, Department of Mathematical Sciences, University of Bath, Claverton Down, Bath, BA2 7AY, United Kingdom**

Key words: Robin boundary condition, stochastic modelling, equivalence framework.

Abstract

Robin (or convective) boundary conditions are employed across the biological and physical sciences, for example, when considering heat conduction or a partially absorbing surface. Moreover, the behaviour of many real-world phenomena are driven by domain growth, in particular, during pattern formation. Due to advances in computing power, stochastic methods have become more popular ways of specifying model systems for such processes in recent years. Here, we present an equivalence framework between a partial differential equation with a general Robin boundary condition on a uniformly growing domain, and two different stochastic representations, referred to as the mesoscale and microscale. We derive local rules which ensure that the stochastic representations match the appropriate PDE boundary condition, and demonstrate through several test problems that theses equivalences are accurate.

1 Introduction

General Robin boundary conditions have applications across the biological and physical sciences. They relate the flux of some entity over a boundary to its concentration at the same point, and are traditionally employed in the context of continuum partial differential equation (PDE) representations. In the context of reaction-diffusion systems, the Robin condition corresponds to a partially absorbing boundary with a constant flux of particles, and when considered in the context of heat conduction, this boundary condition is called a convective boundary condition (Hahn and Özisik, 2012). Due to the need for more accurate and realistic models, stochastic methods are becoming more common, aided by recent advances in computing power. As a result, we need to find equivalent specifications of the Robin boundary condition for such stochastic systems.

Another important consideration when modelling certain real-world phenomena is the influence of domain growth (Deeming and Ferguson, 1990; Gerlee, 2013; Korsgaard and Andersen, 1985; Kulesa et al., 1996; Leshem et al., 1991; Murray et al., 1990). There are many examples of systems that need to be modelled on a growing domain in order to capture the correct behaviour. As an example, consider reaction-diffusion systems. There exist a wealth of biological applications for reaction-diffusion systems on growing domains. Examples range from the movement and proliferation of neural crest cells (Kulesa et al., 2010; McLennan et al., 2012) and the formation of patterns (Crampin et al., 1999; Mort et al., 2016; Woolley et al., 2011).

In this paper, we are interested in how the density of a diffusing species, denoted by $u(x,t)$, evolves at position x for $x \in (0, L(t)) \subseteq \mathbb{R}_{\geq 0}$ and $t > 0$. The spatio-temporal evolution of the density, $u(x,t)$, is governed by the diffusion PDE on a growing domain:

$$\frac{\partial u}{\partial t}(x,t) = D\frac{\partial^2 u}{\partial x^2}(x,t) - \frac{\partial}{\partial x}\left(x\frac{\alpha'(t)}{\alpha(t)}u(x,t)\right), \tag{1}$$

where the domain grows according to $L(t) = L_0\alpha(t)$ with $\alpha(t)$ specifying the rate of uniform domin growth, and D is the Fickian diffusion coefficient. We consider the following general Robin boundary condition at $x = 0$:

$$D\frac{\partial u}{\partial x}(0,t) = \kappa_1 u(0,t) - \kappa_2, \tag{2}$$

for some $\kappa_1, \kappa_2 > 0$. We typically refer to this PDE representation of density as the macroscale, in contrast to the two stochastic representations which we focus on in this paper, which will be known as the mesoscale and microscale respectively.

At the mesoscale, we compartmentalise the spatial domain into compartments of width h, and define the state of the system to be the number of particles in each of these compartments at any given time. Particles may jump between these compartments in order to mimic diffusion, which they do at a rate $d = D/h^2$, where D is the same Fickian diffusion coefficient as in equation (1). There are many simulation techniques for the mesoscale, the most common of which is the event-driven Gillespie algorithm (Gillespie, 1977), where the time until the next event is calculated, and then the event to be enacted is chosen with probability proportional to the rate at which it occurs. There are, however, other methods that are able to simulate systems at this scale (Anderson, 2007; Gibson and Bruck, 2000; Gillespie, 2001). Domain growth will be achieved by employing the stretching method (Smith et al., 2019). This method uniformly spreads the mass by utilising overlap regions, which describe how much a post-growth compartment overlaps a pre-growth one. We direct the interested reader to the aforementioned paper for more details.

At the microscale, we track each individual particle's position, which is updated according to an appropriate stochastic differential equation (SDE). One common method for the simulation of a particle's path is the Euler-Maruyama method (see, for example, Kloeden and Platen (2013)). We wish to generate individual-based rules for each of these stochastic methods in order to recover the Robin boundary condition associated with the PDE in the appropriate regime.

The rest of the paper will be set out as follows. In Section 2, we derive an equivalence for the boundary condition using the mesoscopic representation, while Section 3 is concerned with repeating the process, but for the microscale. In Section 4, we test the equivalence on several test problems, and finish with some concluding remarks in Section 5.

2 Mesoscale

We wish to establish an equivalence between the boundary condition (2) and a second method of modelling diffusion and growth, namely a stochastic, Markovian representation. We discretise the spatial domain into compartments of size h, and allow particles to jump and redistribute between them. In order to establish equivalence of the boundary conditions, we will investigate the evolution of the mean number of particles in the boundary compartment. As such, we define $m_i^k(t)$ to be the number of particles in compartment i on average at time t, when the domain is made up of k compartments in total.

We will begin our analysis by considering the potential changes in the system over an infinitesimally small time interval $[t, t + \delta t)$. The dynamics of the system are comprised of four processes. Firstly, particles may jump to neighbouring compartments with probability $D\delta t/h^2$, where h is the width of a compartment, and we assume that δt is small. Secondly, compartments are added to the domain, mimicking domain growth with probability $kr(k)\delta t$ when there are currently k compartments in total. Here, $r : \mathbb{N} \to \mathbb{R}$ is a function which relates the discretised growth mechanism to $\alpha(t)$ (see Appendix B for more information). We will be utilising the growth mechanism of Smith et al. (2019) in order to conduct the domain growth. This method employs pre/post growth overlap regions whose sizes are defined to be:

$$\delta_i^k = \frac{k+1-i}{k+1}.$$

This δ_i^k is the proportion of post-growth compartment i which overlaps with pre-growth compartment i.

Thirdly, in order to capture partial absorption at the left-hand boundary, we follow (Erban and Chapman, 2007a; Taylor et al., 2015) and let $Q_1 h$ be the probability that a particle is absorbed if it jumps left out of the left-most compartment. Fourthly, we include the contribution of the constant influx at the left-hand boundary by adding particles into the left-most compartment with rate $\frac{DQ_2}{h}\delta t$. We further assume that the probability of two events (diffusion, growth or boundary influx or efflux)

occurring in that small time-step $[t, t+\delta t)$ is $O(\delta t^2)$, and hence disappears when $\delta t \to 0$. Then, we can write:

$$\begin{aligned} m_1^k(t+\delta t) = m_1^k(t) - \frac{2D\delta t}{h^2} m_1^k(t) + \frac{D\delta t}{h^2} m_2^k(t) + (1 - Q_1 h)\frac{D\delta t}{h^2} m_1^k(t) + \frac{DQ_2}{h}\delta t \\ + (k-1)r(k-1)\delta_1^{k-1} m_1^{k-1}(t)\delta t - kr(k) m_1^k(t)\delta t + O(\delta t^2). \end{aligned} \tag{3}$$

The terms on the right hand side can be interpreted as follows, from left to right: the previous state of the system; jumps from compartment one both left and right; jumps into the first compartment from the second; particles reflected back if they have jumped left; influx of particles; gain of particles through the growth mechanism as the domain grows from length $(k-1)$ to k; loss of particles through growth from size k to $k+1$. Subtracting $m_1^k(t)$ from both sides and dividing by δt yields:

$$\frac{m_1^k(t+\delta t) - m_1^k(t)}{\delta t} = \frac{D}{h}\left[\frac{m_2^k(t) - m_1^k(t)}{h} - Q_1 m_1^k(t) + Q_2\right] + \delta_1^{k-1}(k-1)r(k-1)m_1^{k-1}(t) - kr(k)m_1^k(t) + O(\delta t).$$

We need to make a further assumption on how the domain grows, in order to close the system. We assume that the domain grows uniformly, and that mass is spread out evenly, which suggests the following simplifying assumption:

$$(k-1)r(k-1)m_i^{k-1}(t) = kr(k)m_i^k(t), \ i \in \{1, ..., k-1\}. \tag{4}$$

We apply this, and also multiply the equation by $\sqrt{\delta t}$ to obtain

$$\sqrt{\delta t}\left[\frac{m_1^k(t+\delta t) - m_1^k(t)}{\delta t}\right] = \frac{D\sqrt{\delta t}}{h}\left[\frac{m_2^k(t) - m_1^k(t)}{h} - Q_1 m_1^k(t) + Q_2\right] - \sqrt{\delta t}km_1^k(t)\left[r(k) - r(k-1)\delta_1^{k-1}\right] + O(\delta t^{3/2}). \tag{5}$$

We will investigate the final term of the right-hand side of equation (5). Under the above assumptions:

$$\sqrt{\delta t}km_1^k(t)\left[r(k) - r(k-1)\delta_1^{k-1}\right] = \sqrt{\delta t}(k-1)r(k-1)m_1^k(t)\left(\frac{kr(k)}{(k-1)r(k-1)} - \frac{k}{k-1}\delta_1^{k-1}\right).$$

Next, for reasonalble biologically motivated choices of domain growth (such as exponential, linear or logistic, see Appendix B), as the number of compartments gets large, the ratio between the rate of stretching with k and $k-1$ compartments becomes close to 1, and they therefore approach the same value, say r^* (so that $kr(k) \approx (k-1)r(k-1) \approx r^*$). Therefore:

$$\sqrt{\delta t}(k-1)r(k-1)m_1^k(t)\left(\frac{kr(k)}{(k-1)r(k-1)} - \frac{k}{k-1}\delta_1^{k-1}\right) \approx \sqrt{\delta t}r^* m_1^k(t)\left(1 - \frac{k}{k-1}\delta_1^{k-1}\right), \tag{6}$$

which holds for large k. Substituting this into equation (5), we will now take the diffusive limit $\delta t, h \to 0$ (and consequently $k \to \infty$), such that the ratio $\sqrt{\delta t}/h = c$ remains constant. We also note that, from the definition of δ_1^{k-1}, we find that the right-hand side of equation (6) is 0:

$$\delta_1^{k-1} = \frac{k-1}{k} \implies 1 - \frac{k}{k-1}\delta_1^{k-1} = 0.$$

Taking these limits and substituting into equation (5) yields

$$0 = Dc\left[\frac{\partial u}{\partial x}(0,t) - Q_1 u(0,t) + Q_2\right] - 0. \tag{7}$$

Then, comparing the boundary condition (2) to equation (7), we find that

$$D\frac{\partial u}{\partial x}(0,t) - \kappa_1 u(0,t) + \kappa_2 = D\frac{\partial u}{\partial x}(0,t) - DQ_1 u(0,t) + DQ_2.$$

Comparing coefficients we find that $\kappa_1 = DQ_1$ and $\kappa_2 = DQ_2$. We should therefore choose the probability of absorption to be $Q_1 h = \kappa_1 h/D$ and the the rate of influx, Q_2, should be κ_2/D. Note that, because $Q_1 h$ is a probability, this places a constraint on the parameters, namely:

$$\frac{\kappa_1 h}{D} < 1. \tag{8}$$

Intuitively, this says that in order to achieve adsorption at the required rate, for a given diffusion coefficient, the box size must be sufficiently small.

3 Microscale

We now move on to the second stochastic representation, the microscale. Under this modelling methodology, the locations of individual partcles are tracked. Using the Fokker-Planck equation (Risken, 1996), it can be shown that the stochastic differential equation (SDE) for the position of a single particle

$$dX(t) = X(t)\frac{\alpha'(t)}{\alpha(t)}\,dt + \sqrt{2D}\,dW(t) \tag{9}$$

corresponds to the PDE (1) for the probability density of finding a particle at position x at time t. In the SDE (9), $X(t)$ is the location of the particle at time t, $\alpha(t)$ and D are the growth rate and diffusion coefficient respectively (as in PDE (1)) and $dW(t)$ is a standard Weiner process. In this section, we detail how the boundary condition (2) can be simulated under the microscopic modelling paradigm. We follow a similar approach to that of (Singer et al., 2008).

The SDE (9) can be discretised using the Euler-Maruyama method, which gives us a simple update rule for the position of the particle at time $t + \delta t$ based on the position of the particle at time t. Suppose $x_{\delta t}(t)$ is the location of the particle at time t on a discretised time-mesh of size δt. Then the Euler-Maruyama method says:

$$x_{\delta t}(t+\delta t) = x_{\delta t}(t) + x_{\delta t}(t)\frac{\alpha'(t)}{\alpha(t)}\,\delta t + \sqrt{2D\delta t}\xi, \tag{10}$$

where ξ is a normally distributed random variable with zero mean and unit variance.

In order to derive the contributions of the two parts of the boundary condition ($\kappa_1 u(0,t)$ and κ_2), similarly to Section 2, we define probabilities which are governed by individual rules. Firstly, we define the probability related to the adsorption parameter κ_1. In an analogous way to the previous section, we will define the probability of absorbing given that a particle has crossed the boundary in a time-frame $(t, t+\delta t)$. As observed in (Singer et al., 2008), there is a boundary layer of size $\sqrt{\delta t}$ adjacent to the boundary. Within this boundary layer, we are required to define a probability of absorption given that a particle has crossed the boundary. Since particles are only able to cross the boundary in one direction, we require a unidirectional flux, which for Brownian particles, is $O(1/\sqrt{\delta t})$ (Singer and Schuss, 2005). Therefore, the number of particles and consequently the number of particles absorbed must also be $O(1/\sqrt{\delta t})$. In order to ensure that the probability remains constant, we need to multiply by $\sqrt{\delta t}$, and hence, we set the probability of absorption to be $P_1\sqrt{\delta t}$, for some $P_1 > 0$ to be determined in terms of the simulation parameters. For the influx part of the boundary condition related to the parameter κ_2, we will add particles into the domain at the boundary in the time interval $[t, t+\delta t)$ with probability $2P_2\delta t$. The factor of 2 here reflects the fact that if a particle is to be added exactly at 0, half of its mass would

lie outside of $(0,\infty)$, and hence, doubling this contribution we obtain a single particle being initialised at the boundary.

We return to the Euler-Maruyama update equation (10). Using the boundary rules set out above, we have the following possibilities:

$$x_{\delta t}(t+\delta t) = \begin{cases} x_{\delta t}(t) + x_{\delta t}(t)\frac{\alpha'(t)}{\alpha(t)}\ \delta t + \sqrt{2D\delta t}\xi, & \text{if } x_{\delta t}(t) + x_{\delta t}(t)\frac{\alpha'(t)}{\alpha(t)}\ \delta t + \sqrt{2D\delta t}\xi > 0, \\ -x_{\delta t}(t) - x_{\delta t}(t)\frac{\alpha'(t)}{\alpha(t)}\ \delta t - \sqrt{2D\delta t}\xi, & \text{if } x_{\delta t}(t) + x_{\delta t}(t)\frac{\alpha'(t)}{\alpha(t)}\ \delta t + \sqrt{2D\delta t}\xi < 0 \text{ with probability } 1 - P_1\sqrt{\delta t}, \\ \text{absorbed}, & \text{if } x_{\delta t}(t) + x_{\delta t}(t)\frac{\alpha'(t)}{\alpha(t)}\ \delta t + \sqrt{2D\delta t}\xi < 0 \text{ with probability } P_1\sqrt{\delta t}. \end{cases}$$

We wish to relate the Euler-Maruyama method (10) to the PDE (1), which initially requires the definition of the probability density function (pdf) for the location of a particle. Define $p_{\delta t}(y,t|z,s)$ to be the probability that a particle is located at a position y at time t given that at time $s<t$ it was at position z. Then the update rule for this probability is as follows, where we write $p_{\delta t}(y,t) = p_{\delta t}(y,t|z,s)$ for ease of notation:

$$\begin{aligned} p_{\delta t}(y,t+\delta t) = \int_0^\infty \frac{p_{\delta t}(x,t)}{\sqrt{4\pi D\delta t}} \Bigg[& \exp\left\{ -\frac{\left(y - x - x\frac{\alpha'(t)}{\alpha(t)}\delta t\right)^2}{4D\delta t} \right\} \\ & + (1 - P_1\sqrt{\delta t}) \exp\left\{ -\frac{\left(y + x + x\frac{\alpha'(t)}{\alpha(t)}\delta t\right)^2}{4D\delta t} \right\} \Bigg]\ \mathrm{d}x + 2P_2\delta t\delta(y). \end{aligned} \tag{11}$$

The terms on the right hand side are as follows: the first denotes the probability density function that the particle travelled from $x>0$ to $y>0$; the second term is the probability the particle travelled from $x>0$ to $-y<0$ and is reflected back into the domain to position $y>0$; and the third term is the creation of a particle located at $y=0$ with probability $2P_2\delta t$, where $\delta(y)$ is the Dirac delta function.

The rest of this section evaluates the integral equation (11) in order to calculate the values of P_1 and P_2 in terms of other simulation parameters. In order to do this, we need to solve both within the boundary layer (henceforth called the "inner solution", see Section 3.1) and throughout the rest of the domain (called the "outer solution", Section 3.2). We match these inner and outer solutions in Section 3.2 in order to calculate P_1 and P_2 explicitly.

3.1 Inner solution

We begin by looking at the inner solution. In order to do this, we will employ a change of variables from (x,y,t) to (χ,η,t):

$$\begin{aligned} x &= \chi\sqrt{\delta t}, \\ y &= \eta\sqrt{\delta t}. \end{aligned} \tag{12}$$

Substituting the transformation (12) into equation (11), and defining $p_{\delta t}(\eta\sqrt{\delta t},t) := p_{\text{in}}(\eta,t)$ and $p_{\delta t}(\chi\sqrt{\delta t},t) := p_{\text{in}}(\chi,t)$ yields the integral form using inner coordinates:

$$\begin{aligned} p_{\text{in}}(\eta,t+\delta t) = \int_0^\infty \frac{p_{\text{in}}(\chi,t)}{\sqrt{4\pi D}} \Bigg[& \exp\left\{ -\frac{\left(\eta - \chi - \chi\frac{\alpha'(t)}{\alpha(t)}\delta t\right)^2}{4D} \right\} \\ & + (1 - P_1\sqrt{\delta t}) \exp\left\{ -\frac{\left(\eta + \chi + \chi\frac{\alpha'(t)}{\alpha(t)}\delta t\right)^2}{4D} \right\} \Bigg]\ \mathrm{d}\chi + 2P_2\delta t\delta(\eta\sqrt{\delta t}). \end{aligned} \tag{13}$$

We now assume that there is an expansion of p_{in} in powers of $\sqrt{\delta t}$. This means that we have, up to first order in $\sqrt{\delta t}$:

$$\begin{aligned} p_{\text{in}}(\eta, t+\delta t) &\approx p_{\text{in}}(\eta, t) + \delta t \frac{\partial p_{\text{in}}}{\partial t}(\eta, t) \approx p_{\text{in}}^{(0)}(\eta, t) + \sqrt{\delta t} p_{\text{in}}^{(1)}(\eta, t) + O(\delta t), \\ p_{\text{in}}(\chi, t) &\approx p_{\text{in}}^{(0)}(\chi, t) + \sqrt{\delta t} p_{\text{in}}^{(1)}(\chi, t). \end{aligned} \tag{14}$$

We will further define the function $f : \mathbb{R}^3_{\geq 0} \to \mathbb{R}_{\geq 0}$ to be

$$f(\chi, \eta, u) = \exp\left\{ -\frac{\left(\eta - \chi - \chi \frac{\alpha'(t)}{\alpha(t)} u^2\right)^2}{4D} \right\} + (1 - P_1 u) \exp\left\{ -\frac{\left(\eta + \chi + \chi \frac{\alpha'(t)}{\alpha(t)} u^2\right)^2}{4D} \right\},$$

so that the integral equation (13) may be written as

$$p_{\text{in}}^{(0)}(\eta, t) + \sqrt{\delta t} p_{\text{in}}^{(1)}(\eta, t) \approx \int_0^\infty \frac{p_{\text{in}}^{(0)}(\chi, t) + \sqrt{\delta t} p_{\text{in}}^{(1)}(\chi, t)}{\sqrt{4\pi D}} f(\chi, \eta, \sqrt{\delta t}) \, \mathrm{d}\chi + 2P_2 \sqrt{dt} \delta(\eta), \tag{15}$$

where we have also utilised the scaling property of the Dirac delta, $\delta(ax) = \frac{1}{|a|}\delta(x)$, to simplify the final term on the right-hand side.

In order to compare terms at different orders in $\sqrt{\delta t}$, we will Taylor expand the function f about $u = 0$, which yields:

$$\begin{aligned} f(\chi, \eta, u) &\approx f(\chi, \eta, 0) + u f_u(\chi, \eta, 0) \\ &\approx \exp\left\{ -\frac{(\eta - \chi)^2}{4D} \right\} + \exp\left\{ -\frac{(\eta + \chi)^2}{4D} \right\} - P_1 u \exp\left\{ -\frac{(\eta + \chi)^2}{4D} \right\}, \\ \implies f(\chi, \eta, \sqrt{\delta t}) &\approx \exp\left\{ -\frac{(\eta - \chi)^2}{4D} \right\} + \exp\left\{ -\frac{(\eta + \chi)^2}{4D} \right\} - P_1 \sqrt{\delta t} \exp\left\{ -\frac{(\eta + \chi)^2}{4D} \right\}. \end{aligned} \tag{16}$$

Substituting equation (16) into (15) and collecting terms of the same order in $\sqrt{\delta t}$, we obtain the following expansion:

$$\begin{aligned} p_{\text{in}}^{(0)}(\eta, t) + \sqrt{\delta t} p_{\text{in}}^{(1)}(\eta, t) \approx{}& \frac{1}{\sqrt{4\pi D}} \int_0^\infty p_{\text{in}}^{(0)}(\chi, t) \left[\exp\left\{ -\frac{(\eta - \chi)^2}{4D} \right\} + \exp\left\{ -\frac{(\eta + \chi)^2}{4D} \right\} \right] \, \mathrm{d}\chi \\ &+ \frac{\sqrt{\delta t}}{\sqrt{4\pi D}} \int_0^\infty p_{\text{in}}^{(1)}(\chi, t) \left[\exp\left\{ -\frac{(\eta - \chi)^2}{4D} \right\} + \exp\left\{ -\frac{(\eta + \chi)^2}{4D} \right\} \right] \\ &- P_1 p_{\text{in}}^{(0)}(\chi, t) \exp\left\{ -\frac{(\eta + \chi)^2}{4D} \right\} \, \mathrm{d}\chi + 2P_2 \sqrt{\delta t} \delta(\eta). \end{aligned} \tag{17}$$

For simplicity, we will define a function $K : \mathbb{R}^2_{\geq 0} \to \mathbb{R}_{\geq 0}$:

$$K(\chi, \eta) = \exp\left\{ -\frac{(\eta - \chi)^2}{4D} \right\} + \exp\left\{ -\frac{(\eta + \chi)^2}{4D} \right\}. \tag{18}$$

Then, comparing powers of $\sqrt{\delta t}$ in equation (17), we find that:

$$p_{\text{in}}^{(0)}(\eta, t) = \frac{1}{\sqrt{4\pi D}} \int_0^\infty p_{\text{in}}^{(0)}(\chi, t) K(\chi, \eta) \, \mathrm{d}\chi, \tag{19}$$

$$p_{\text{in}}^{(1)}(\eta, t) = \frac{1}{\sqrt{4\pi D}} \int_0^\infty p_{\text{in}}^{(1)}(\chi, t) K(\chi, \eta) - P_1 p_{\text{in}}^{(0)}(\chi, t) \exp\left\{ -\frac{(\eta + \chi)^2}{4D} \right\} \, \mathrm{d}\chi + 2P_2 \delta(\eta). \tag{20}$$

We simplify equation (19) by firstly noting that K is even in χ, so that $K(\chi,\eta) = K(-\chi,\eta)$. Therefore, extending p_{in} to be an even function in its first variable, we can extend the lower bound of the integral to $-\infty$. This doubles the right-hand side, but we are able to halve the integral by only taking the first term in K (each of the two terms contributes the same amount when integrated over the whole of the real line). Therefore:

$$p_{\text{in}}^{(0)}(\eta,t) = \int_{-\infty}^{\infty} p_{\text{in}}^{(0)}(\chi,t)\frac{1}{\sqrt{4\pi D}}\exp\left\{-\frac{(\eta-\chi)^2}{4D}\right\}\,\mathrm{d}\chi.$$

We recognise the RHS as a convolution, and so, if we take the Fourier transform of both sides, we obtain:

$$\hat{p}_{\text{in}}^{(0)}(\omega,t) = \hat{p}_{\text{in}}^{(0)}(\omega,t)\exp\left\{-D\omega^2\right\}, \tag{21}$$

where $\hat{p}_{\text{in}}^{(0)}(\omega,t)$ denotes the Fourier transform of $p_{\text{in}}^{(0)}(\eta,t)$, and we have employed the convolution theorem on the right-hand side. In order to satisfy equation (21), for any value of $\omega \neq 0$, we must have that $\hat{p}_{\text{in}}^{(0)}(\omega,t) = 0$, however, at $\omega = 0$, $\hat{p}_{\text{in}}^{(0)}(\omega,t)$ the value of $\hat{p}_{\text{in}}^{(0)}(0,t)$ is not specified, however we know that the inverse Fourier transform must be non-negative. As such, $\hat{p}_{\text{in}}^{(0)}(\omega,t) = \delta(\omega)$. Employing the inverse Fourier transform tells us that $p_{\text{in}}^{(0)}(\eta,t)$ is constant, and so, without loss of generality, we set

$$p_{\text{in}}^{(0)}(\eta,t) = p_{\text{in}}^{(0)}(0,t), \qquad \forall \eta > 0. \tag{22}$$

We now investigate equation (20). We begin by simplifying this equation by substituting in the result from equation (22), and evaluating the second term in the integral, to obtain:

$$p_{\text{in}}^{(1)}(\eta,t) = \frac{1}{\sqrt{4\pi D}}\int_0^{\infty} p_{\text{in}}^{(1)}(\chi,t)K(\chi,\eta)\,\mathrm{d}\chi - \frac{P_1}{2}p_{\text{in}}^{(0)}(0,t)\mathrm{erfc}\left(\frac{\eta}{\sqrt{4D}}\right) + 2P_2\delta(\eta).$$

Taking the derivative with respect to η gives the following:

$$\frac{\partial p_{\text{in}}^{(1)}}{\partial\eta}(\eta,t) = \frac{1}{\sqrt{4\pi D}}\int_0^{\infty} p_{\text{in}}^{(1)}(\chi,t)\frac{\partial K}{\partial\eta}(\chi,\eta)\,\mathrm{d}\chi + P_1 p_{\text{in}}^{(0)}(0,t)\frac{1}{\sqrt{4\pi D}}\exp\left\{-\frac{\eta^2}{4D}\right\} + 2P_2\delta'(\eta). \tag{23}$$

We wish to rewrite this equation to have the derivative of $p_{\text{in}}^{(1)}$ on both sides in order to obtain an implicit equation on which we can later use the Fourier transform. To do this, we note that if we let $\tilde{K} : \mathbb{R}^2_{\geq 0} \to \mathbb{R}_{\geq 0}$ be defined by

$$\begin{aligned}\tilde{K}(\chi,\eta) &= -\exp\left\{-\frac{(\eta-\chi)^2}{4D}\right\} + \exp\left\{-\frac{(\eta+\chi)^2}{4D}\right\}\\ \text{then } \frac{\partial K}{\partial\eta}(\chi,\eta) &= \frac{\partial\tilde{K}}{\partial\chi}(\chi,\eta).\end{aligned}$$

Applying this relationship to equation (23) and using integration by parts to move the derivative from $\tilde{K}$ onto $p_{\text{in}}^{(1)}$, we obtain:

$$\frac{\partial p_{\text{in}}^{(1)}}{\partial\eta}(\eta,t) = -\frac{1}{\sqrt{4\pi D}}\int_0^{\infty} \frac{\partial p_{\text{in}}^{(1)}}{\partial\chi}(\chi,t)\tilde{K}(\chi,\eta)\,\mathrm{d}\chi + P_1 p_{\text{in}}^{(0)}(0,t)\frac{1}{\sqrt{4\pi D}}\exp\left\{-\frac{\eta^2}{4D}\right\} + 2P_2\delta'(\eta). \tag{24}$$

We now introduce two functions which will be used in order to simplify equation (24). Let $h, \phi : \mathbb{R}^2_{\geq 0} \to \mathbb{R}$ be defined by:

$$h(\eta,t) = \frac{\partial p_{\text{in}}^{(1)}}{\partial\eta}(\eta,t) - P_1 p_{\text{in}}^{(0)}(0,t)\frac{1}{\sqrt{4\pi D}}\exp\left\{-\frac{\eta^2}{4D}\right\}, \tag{25}$$

$$\phi(\eta,t) = 2P_2\delta'(\eta) - P_1 p_{\text{in}}^{(0)}(0,t)\frac{1}{\sqrt{4\pi D}}\int_0^\infty \exp\left\{-\frac{\eta^2}{4D}\right\}\tilde{K}(\chi,\eta)\ \mathrm{d}\chi, \tag{26}$$

then equation (24) may be written as:

$$h(\eta,t) = -\frac{1}{\sqrt{4\pi D}}\int_0^\infty h(\chi,t)\tilde{K}(\chi,\eta)\ \mathrm{d}\chi + \phi(\eta,t). \tag{27}$$

The goal here is to calculate $h(\eta,t)$. Once this has been found, we will then obtain a representation for the derivative of $p_{\text{in}}^{(1)}$. In order to find h, we recognise that the integral on the right hand side of equation (27) is of a similar form to that of equation (19), and we will therefore employ a similar technique. On this occasion, note that $\tilde{K}$ is odd in χ, and hence, if we extend h to be an odd function in the first variable, we obtain

$$h(\eta,t) = \frac{1}{\sqrt{4\pi D}}\int_{-\infty}^\infty h(\chi,t)\exp\left\{-\frac{(\eta-\chi)^2}{4D}\right\}\ \mathrm{d}\chi + \phi(\eta,t),$$

and once again recognise this integral as a convolution. Taking the Fourier transform of both sides and employing the convolution theorem yields

$$\hat{h}(\omega,t) = \hat{h}(\omega,t)\exp\left\{-D\omega^2\right\} + \hat{\phi}(\omega,t) \quad\Longrightarrow\quad \hat{h}(\omega,t) = \frac{\hat{\phi}(\omega,t)}{1-\exp\left\{D\omega^2\right\}}, \tag{28}$$

where $\hat{h}$ and $\hat{\phi}$ are the respective Fourier transforms. We therefore need to find the Fourier transform of ϕ. From equation (26), we can explicitly calculate the integral and find that

$$\phi(\eta,t) = 2P_2\delta'(\eta) + P_1 p_{\text{in}}^{(0)}(0,t)\frac{1}{\sqrt{8\pi D}}\exp\left\{-\frac{\eta^2}{8D}\right\}\operatorname{erf}\left(\frac{\eta}{\sqrt{8D}}\right). \tag{29}$$

We cannot directly find the Fourier transform of the function (29). As a result, we will take a Taylor expansion of the Fourier transform:

$$\hat{\phi}(\omega,t) \approx \hat{\phi}(0,t) + \hat{\phi}_\omega(0,t) + \frac{1}{2}\hat{\phi}_{\omega\omega}(0,t), \tag{30}$$

where the subscripts indicate derivatives. We simplify each of the three terms in equation (30) in turn.

$\underline{\hat{\phi}(0,t)}$

By the definition of the Fourier transform, we have that

$$\begin{aligned}
\hat{\phi}(0,t) &= \left(\int_{-\infty}^\infty \phi(\eta,t)e^{-i\omega\eta}\ \mathrm{d}\eta\right)\Big|_{\omega=0} \\
&= \int_{-\infty}^\infty \phi(\eta,t)\ \mathrm{d}\eta \\
&= 2P_2\int_{-\infty}^\infty \delta'(\eta)\ \mathrm{d}\eta + P_1 p_{\text{in}}^{(0)}(0,t)\frac{1}{\sqrt{4\pi D}}\int_{-\infty}^\infty \exp\left\{-\frac{\eta^2}{8D}\right\}\operatorname{erf}\left(\frac{\eta}{\sqrt{8D}}\right)\ \mathrm{d}\eta.
\end{aligned}$$

The first integral evaluates to zero. The second integral is also 0 because the integrand is odd. Therefore:

$$\hat{\phi}(0,t) = 0. \tag{31}$$

$\underline{\hat{\phi}_\omega(0,t)}$

In order to simplify the second term on the right hand side of the Taylor expansion given in equation (30), we employ the following relationship for derivatives of the Fourier transform:

$$\hat{\phi}_\omega(\omega,t) = i\int_{-\infty}^{\infty} \eta\phi(\eta,t)e^{-i\omega\eta}\ \mathrm{d}\eta. \tag{32}$$

We substitute expression (29) for $\phi(\eta,t)$ and set $\omega = 0$ in equation (32) and simplify:

$$\begin{aligned}\hat{\phi}_\omega(0,t) &= iP_1 p_{\text{in}}^{(0)}(0,t)\frac{1}{\sqrt{8\pi D}}\int_{-\infty}^{\infty}\eta\exp\left\{-\frac{\eta^2}{8D}\right\}\mathrm{erf}\left(\frac{\eta}{\sqrt{8D}}\right)\ \mathrm{d}\eta + 2iP_2\int_{-\infty}^{\infty}\eta\delta'(\eta)\ \mathrm{d}\eta \\ &= 2iP_1 p_{\text{in}}^{(0)}(0,t)\sqrt{\frac{D}{\pi}} - 2iP_2.\end{aligned} \tag{33}$$

$\underline{\hat{\phi}_{\omega\omega}(0,t)}$

We will again make use of a formula that relates the second derivative of the Fourier transform to the function itself:

$$\hat{\phi}_{\omega\omega}(\omega,t) = -\int_{-\infty}^{\infty}\eta^2\phi(\eta,t)e^{-i\omega\eta}\ \mathrm{d}\eta. \tag{34}$$

Setting $\omega = 0$ in equation (34) and substituting in for $\phi(\eta,t)$ from equation (29) yields:

$$\hat{\phi}_{\omega\omega}(0,t) = -P_1 p_{\text{in}}^{(0)}(0,t)\frac{1}{\sqrt{4\pi D}}\int_{-\infty}^{\infty}\eta^2\exp\left\{-\frac{\eta^2}{8D}\right\}\mathrm{erf}\left(\frac{\eta}{\sqrt{8D}}\right)\ \mathrm{d}\eta - 2P_2\int_{-\infty}^{\infty}\eta^2\delta'(\eta)\ \mathrm{d}\eta = 0. \tag{35}$$

Now that we have expressions for each of the terms in equation (30) We now substitute equations (31), (33) and (35) into the Taylor expansion to obtain, for small ω:

$$\hat{\phi}(\omega,t) \approx 2i\left[P_1 p_{\text{in}}^{(0)}(0,t)\sqrt{\frac{D}{\pi}} - P_2\right]\omega, \tag{36}$$

which, when substituted into equation (37) yields the following expression for the Fourier transform of h for small ω:

$$\hat{h}(\omega,t) = \frac{2i\left[P_1 p_{\text{in}}^{(0)}(0,t)\sqrt{\frac{D}{\pi}} - P_2\right]\omega}{1-\exp\left\{-D\omega^2\right\}}, \tag{37}$$

which we now invert.

In order to invert $\hat{h}$, we firstly note that the poles of the function are symmetric in the complex plane about the real axis. This means that for every pole, there is another pole which is its complex conjugate. We apply a Weiner-Hopf decomposition to $h(\eta,t)$ by writing

$$\begin{aligned}h_+(\eta,t) &= h(\eta,t)\mathbb{1}_{[\eta>0]}, \\ h_-(\eta,t) &= h(\eta,t)\mathbb{1}_{[\eta<0]},\end{aligned}$$

where $\mathbb{1}_{[A]}$ is the indicator function that takes the value 1 if the event A is true, and the value 0 otherwise. Using this decomposition we can write the Fourier transform of h as

$$\hat{h}(\omega,t) = \hat{h}_+(\omega,t) + \hat{h}_-(\omega,t),$$

where $\hat{h}_\pm(\omega,t)$ are the Fourier transforms of $h_\pm(\eta,t)$. This decomposition allows us to split the poles, so that only those poles of $\hat{h}$ which have positive imaginary part are poles of $\hat{h}_+$, and similarly those poles

with negative imaginary part are poles of $\hat{h}_-$. We also note that there is a simple pole at $\omega = 0$, which is common to both parts of the decomposition. Using this information, we need only to investigate one of $\hat{h}_\pm$. In this case, we choose to focus on $\hat{h}_+(\omega, t)$, and note that it contributes a half to the Fourier transform of h. Therefore

$$\begin{aligned}\hat{h}_+(\omega, t) &= \frac{1}{2}\hat{h}(\omega, t) \\ &\approx \frac{1}{2}\frac{\hat{\phi}(\omega, t)}{1 - \exp\{-D\omega^2\}} \\ &\approx \frac{1}{2}\frac{\hat{\phi}(\omega, t)}{D\omega^2} \\ &= \left[P_1 p_{\text{in}}^{(0)}(0, t)\frac{1}{\sqrt{D\pi}} - \frac{P_2}{D}\right]\frac{i}{\omega},\end{aligned}$$

where, in the third line, we have approximated the denominator as a Taylor series to first order, for small values of ω. We will invert this Fourier transform by closing the contour of integration around the lower half-plane (see Figure 1). Since all of the poles of $\hat{h}_+(\omega, t)$ are either 0 or in the upper half plane, we only need to consider the simple pole at $\omega = 0$. We begin by defining four curves (see figure 1):

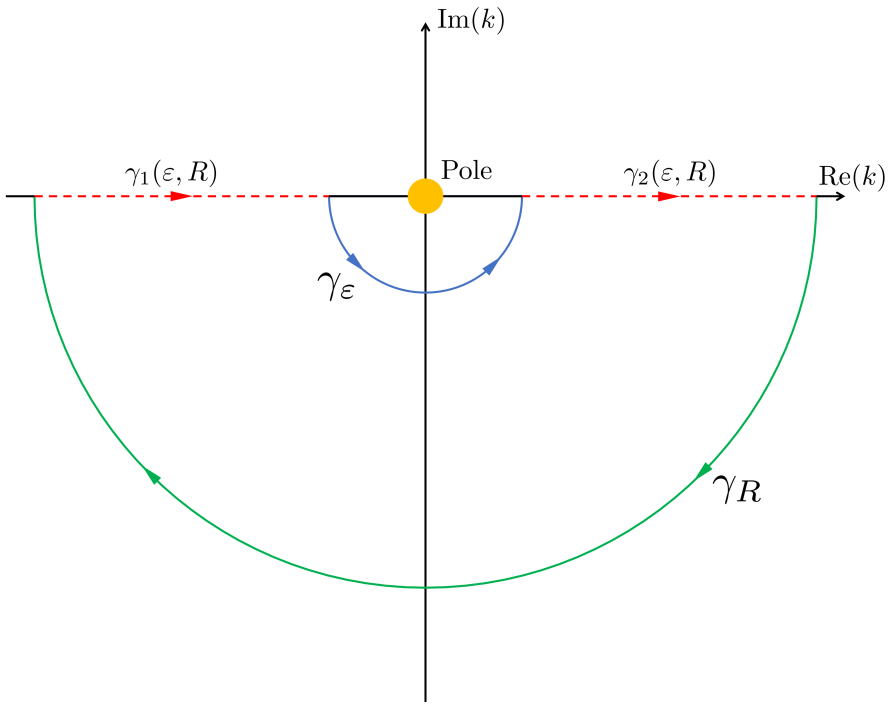


Figure 1. Contour integral about the lower half plane. The green clockwise oriented curve is a semi-circle of radius R, the blue counter-clockwise curve is a semi-circle of radius ε and the two red dashed lines are line segments $(-R, -\varepsilon)$ and (ε, R). The union of these four curves forms $\gamma(\varepsilon, R)$.

- $\gamma_R = \left\{\omega = Re^{i\theta} : \theta \in (-\pi, 0)\right\}$, oriented clockwise,
- $\gamma_\varepsilon = \left\{\omega = \varepsilon e^{i\theta} : \theta \in (-\pi, 0)\right\}$, oriented counter-clockwise,

- $\gamma_1(\varepsilon, R) = (-R, -\varepsilon)$,
- $\gamma_2(\varepsilon, R) = (\varepsilon, R)$.

We further define $\gamma(\varepsilon, R)$ to be the union of the four curves above. Then

$$\oint_{\gamma(\varepsilon,R)} \hat{h}_+(\omega,t)e^{i\omega\eta}\,\mathrm{d}\omega = \int_{\gamma_R} \hat{h}_+(\omega,t)e^{i\omega\eta}\,\mathrm{d}\omega + \int_{\gamma_\varepsilon} \hat{h}_+(\omega,t)e^{i\omega\eta}\,\mathrm{d}\omega + \int_{\gamma_1(\varepsilon,R)} \hat{h}_+(\omega,t)e^{i\omega\eta}\,\mathrm{d}\omega + \int_{\gamma_2(\varepsilon,R)} \hat{h}_+(\omega,t)e^{i\omega\eta}\,\mathrm{d}\omega. \tag{38}$$

We take the limit as $\varepsilon \to 0$ and $R \to \infty$ in equation (38) and re-arrange to obtain

$$\begin{aligned}\int_{-\infty}^{\infty} \hat{h}_+(\omega,t)e^{i\omega\eta}\,\mathrm{d}\omega = \lim_{\varepsilon\to 0}\lim_{R\to\infty}\left(\oint_{\gamma(\varepsilon,R)} \hat{h}_+(\omega,t)e^{i\omega\eta}\,\mathrm{d}\omega\right) - \lim_{R\to\infty}\left(\int_{\gamma_R} \hat{h}_+(\omega,t)e^{i\omega\eta}\,\mathrm{d}\omega\right) \\ - \lim_{\varepsilon\to 0}\left(\int_{\gamma_\varepsilon} \hat{h}_+(\omega,t)e^{i\omega\eta}\,\mathrm{d}\omega\right).\end{aligned} \tag{39}$$

The left hand side of equation (39) is 2π times the inverse Fourier transform of $\hat{h}_+$, which we need to calculate. Terms on the right hand side can be simplified utilising results from complex analysis, namely Cauchy's residue theorem, Jordan's lemma and the indentation lemma respectively.

- The first term on the right hand side is 0 by Cauchy's residue theorem: since the contour $\gamma(\varepsilon, R)$ does not contain a pole of $\hat{h}_+(\omega, t)$ within its interior, the contour integral is 0.
- The second term on the right hand side is simplified using Jordan's lemma. This says that if an integral is of the form $I = \int_C g(z)e^{-iqz}\,\mathrm{d}z$, where $q > 0$ and C is a semi-circular arc of size $R > 0$ in the lower half-plane, then for some upper bound $g_{\max}$ which depends on the contour C,

  if
  $$\forall z \in C, |g(z)| \le g_{\max},$$
  then
  $$I \to 0 \text{ if } g_{\max} \to 0.$$

  Changing variables in the integral of the second term of the right hand side of equation (39) yields $\int_{\gamma_R} \hat{h}_+(\omega,t)e^{i\omega\eta}\,\mathrm{d}\omega = -\int_{\gamma_R} \hat{h}_+(\omega,t)e^{-i\omega\eta}\,\mathrm{d}\omega$. Now, since ω lies on the semi-circle of radius R, we can write, for $\theta \in (-\pi, 0)$:

  $$\begin{aligned}|\hat{h}_+(\omega,t)| &= |\hat{h}_+(Re^{i\theta},t)| \\ &= \left|P_1 p_{\text{in}}^{(0)}(0,t)\frac{1}{\sqrt{D\pi}} - \frac{P_2}{D}\right|\left|\frac{i}{Re^{i\theta}}\right| \\ &= \frac{\mu}{R},\end{aligned}$$

  where $\mu = \left|P_1 p_{\text{in}}^{(0)}(0,t)\frac{1}{\sqrt{D\pi}} - \frac{P_2}{D}\right|$ is constant. Therefore, defining $g_{\max} := \mu/R$, we have $|\hat{h}_+(\omega,t)| \le g_{\max}$. By Jordan's lemma, as $R \to \infty$, $g_{\max} \to 0$ and hence the integral tends to 0.

- The final term is simplified using the indentation lemma. This lemma states that if a contour $C_\varepsilon = \varepsilon e^{i\theta}$ for $\theta \in (\theta_1, \theta_2)$ $(0 \le \theta_1 < \theta_2 \le 2\pi)$ partially surrounds a simple pole at 0 for a function f, then
$$\lim_{\varepsilon \to 0} \left(\int_{C_\varepsilon} f(z)\ \mathrm{d}z \right) = i(\theta_2 - \theta_1) \mathrm{Res}\left[f(z) \right]_{z=0}, \tag{40}$$
where $\mathrm{Res}\left[f(z) \right]_{z=0}$ is the residual of the function f at the pole $z = 0$. In the case of the final term of equation (39), we have $\theta_1 = \pi$, $\theta_2 = 2\pi$ and $\mathrm{Res}\left[\hat{h}_+(\omega, t) e^{i\omega\eta} \right]_{\omega=0} = i\left(P_1 p_{\text{in}}^{(0)}(0,t) \frac{1}{\sqrt{D\pi}} - \frac{P_2}{D} \right)$. Therefore
$$\lim_{\varepsilon \to 0} \left(\int_{-\infty}^{\infty} \hat{h}_+(\omega, t) e^{i\omega\eta} \right) = -\pi \left(P_1 p_{\text{in}}^{(0)}(0,t) \frac{1}{\sqrt{D\pi}} - \frac{P_2}{D} \right).$$

Putting these three elements together into equation (39), we obtain an expression for the function $h(\eta, t)$:
$$\begin{aligned} h(\eta, t) &= \frac{1}{2\pi} \int_{-\infty}^{\infty} \hat{h}(\omega, t) e^{i\omega\eta}\ \mathrm{d}\omega \\ &= \frac{1}{\pi} \int_{-\infty}^{\infty} \hat{h}_+(\omega, t) e^{i\omega\eta}\ \mathrm{d}\omega \\ &= \frac{1}{\pi} \pi \left(P_1 p_{\text{in}}^{(0)}(0,t) \frac{1}{\sqrt{D\pi}} - \frac{P_2}{D} \right) \\ &= P_1 p_{\text{in}}^{(0)}(0,t) \frac{1}{\sqrt{D\pi}} - \frac{P_2}{D}. \end{aligned}$$

Finally, we recall the definition of h from equation (25) to obtain the following expression for the derivative of $p_{\text{in}}^{(1)}(\eta, t)$:
$$\frac{\partial p_{\text{in}}^{(1)}}{\partial \eta}(\eta, t) = P_1 p_{\text{in}}^{(0)}(0,t) \frac{1}{\sqrt{D\pi}} \left(1 + \frac{1}{2} \exp\left\{ -\frac{\eta^2}{4D} \right\} \right) - \frac{P_2}{D}. \tag{41}$$

3.2 Outer solution and matching

In this section, we will investigate the outer solution, and match it to the inner solution. The outer solution solves the PDE away from the boundary layer. We return to the integral equation (11) and note that, when considering the outer solution, $\sqrt{\delta t}$ is small compared to the spatial coordinates. Under this assumption, integral (11) becomes:
$$p_{\text{out}}(y, t+\delta t) = \int_0^\infty \frac{p_{\text{out}}(x,t)}{\sqrt{4\pi D \delta t}} \left[\exp\left\{ -\frac{\left(y - x - x\frac{\alpha'(t)}{\alpha(t)}\delta t \right)^2}{4D\delta t} \right\} + \exp\left\{ -\frac{\left(y + x + x\frac{\alpha'(t)}{\alpha(t)}\delta t \right)^2}{4D\delta t} \right\} \right] \mathrm{d}x,$$
where we have also changed $p_{\delta t}$ to be p_{out} to reflect the fact that we are considering the outer solution. As previously, we recognise the function in the square brackets is even in x, so we extend p_{out} to be an even function in the first variable to obtain the integral form:
$$p_{\text{out}}(y, t+\delta t) = \int_{-\infty}^\infty \frac{p_{\text{out}}(x,t)}{\sqrt{4\pi D \delta t}} \exp\left\{ -\frac{\left(y - x - x\frac{\alpha'(t)}{\alpha(t)}\delta t \right)^2}{4D\delta t} \right\} \mathrm{d}x. \tag{42}$$

We note that this integral equation is the fundamental solution of the equation
$$\frac{p_{\text{out}}(y, t+\delta t) - p_{\text{out}}(y,t)}{\delta t} = D\frac{\partial^2 p_{\text{out}}}{\partial y^2}(y,t) - \frac{\partial}{\partial y}\left(y \frac{\alpha'(t)}{\alpha(t)} p_{\text{out}}(y,t) \right) + O(\sqrt{\delta t}), \tag{43}$$

which is a time-discretised version of the PDE (1). We now use a similar method to that of the inner solution, and expand all functions in equation (42) in powers of $\sqrt{\delta t}$. We therefore write, to first order:

$$\begin{aligned} p_{\text{out}}(y, t+\delta t) &\approx p_{\text{out}}^{(0)}(y,t) + \sqrt{\delta t} p_{\text{out}}^{(1)}(y,t), \\ p_{\text{out}}(x,t) &\approx p_{\text{out}}^{(0)}(x,t) + \sqrt{\delta t} p_{\text{out}}^{(1)}(x,t). \end{aligned} \tag{44}$$

Now that we have established the outer solution and its expansion, we are able to write down the rules which enable us to match this outer solution to the inner solution from the previous section. For this, we employ van Dyke's matching rule. Generally, van Dyke's matching rule means that the outer limit of the inner solution and the inner limit of the outer solution must coincide at all orders of $\sqrt{\delta t}$ in their respective expansions. In our case, this states that:

$$\lim_{\eta\to\infty}\left(p_{\text{in}}^{(0)}(\eta,t)\right) = \lim_{y\to 0}\left(p_{\text{out}}^{(0)}(y,t)\right), \tag{45}$$

$$\lim_{\eta\to\infty}\left(p_{\text{in}}^{(1)}(\eta,t)\right) = \lim_{y\to 0}\left(p_{\text{out}}^{(1)}(y,t)\right). \tag{46}$$

For the zeroth order expansion, $p_{\text{out}}^{(0)}(y,t)$ solves the fundamental solution (43), and as a result, is a continuous function. This means that taking the limit on the right hand side of equation (45) as $y \to 0$ is the same as evaluating at $y = 0$. Using this fact, together with equation (22), we obtain that

$$p_{\text{in}}(\eta, t) = p_{\text{out}}(0,t) \quad \forall \eta > 0. \tag{47}$$

We now need to calculate the first order term $p_{\text{out}}^{(1)}(y,t)$. In order to do so, we will further define a function $g : \mathbb{R}^3_{\geq 0} \to \mathbb{R}_{\geq 0}$ such that:

$$g(x,y,u) = \frac{1}{\sqrt{4\pi D u}} \exp\left\{ -\frac{\left(y - x - x\frac{\alpha'}{\alpha}u^2\right)^2}{4Du^2} \right\}. \tag{48}$$

Substituting equations (44) and (48) into equation (42) to obtain

$$p_{\text{out}}^{(0)}(y,t) + \sqrt{\delta t} p_{\text{out}}^{(1)}(y,t) \approx \int_{-\infty}^{\infty} (p_{\text{out}}^{(0)}(x,t) + \sqrt{\delta t} p_{\text{out}}^{(1)}(x,t)) g(x,y,\sqrt{\delta t})\ \mathrm{d}x.$$

We Taylor expand the function g about $u = 0$ to first order, which when substituted into the integral equation yields:

$$\begin{aligned} p_{\text{out}}^{(0)}(y,t) + \sqrt{\delta t} p_{\text{out}}^{(1)}(y,t) \approx & \int_{-\infty}^{\infty} p_{\text{out}}^{(0)}(x,t) \lim_{u\to 0}\left(g(x,y,u)\right)\ \mathrm{d}x \\ & + \sqrt{\delta t}\int_{-\infty}^{\infty} p_{\text{out}}^{(1)}(x,t) \lim_{u\to 0}\left(g(x,y,u)\right) + p_{\text{out}}^{(0)}(x,t) \lim_{u\to 0}\left(g_u(x,y,u)\right)\ \mathrm{d}x. \end{aligned} \tag{49}$$

Comparing coefficients of $\sqrt{\delta t}$ we obtain:

$$p_{\text{out}}^{(1)}(y,t) = \int_{-\infty}^{\infty} p_{\text{out}}^{(1)}(x,t) \lim_{u\to 0}\left(g(x,y,u)\right) + p_{\text{out}}^{(0)}(x,t) \lim_{u\to 0}\left(g_u(x,y,u)\right)\ \mathrm{d}x. \tag{50}$$

We show in Appendix A that, for large $y = \eta u$:

$$\lim_{u\to 0}\left(g(x,y,u)\right) \sim \delta(y-x),$$

$$\lim_{u\to 0}\left(g_u(x,y,u)\right) \sim \eta\delta'(y-x).$$

Substituting these expressions into the integral equation (50) we obtain that, for large values of η:

$$p^{(1)}_{\text{out}}(y,t) \sim p^{(1)}_{\text{out}}(y,t) + \eta\frac{\partial p^{(0)}_{\text{out}}}{\partial y}(y,t),$$

and hence, by the matching rule:

$$p^{(1)}_{\text{in}}(\eta,t) \sim p^{(1)}_{\text{out}}(0,t) + \eta\frac{\partial p^{(0)}_{\text{out}}}{\partial y}(0,t)$$

for large η — $p^{(1)}_{\text{in}}(\eta,t)$ is asymptotically linear in η. If we differentiate both sides with respect to η:

$$\frac{\partial p^{(1)}_{\text{in}}}{\partial \eta}(\eta,t) = \frac{\partial p^{(0)}_{\text{out}}}{\partial y}(0,t) \quad \text{as } \eta\to\infty. \tag{51}$$

Revisiting the inner solution, and equation (41), we can relate $\frac{\partial p^{(0)}_{\text{out}}}{\partial \eta}$ to $p^{(0)}_{\text{out}}$. We recall that $p^{(0)}_{\text{out}}$ is the solution of the PDE (1), and hence this will allow us to relate our individual parameters with the boundary condition (2):

$$\begin{aligned}
\frac{\partial p^{(0)}_{\text{out}}}{\partial y}(0,t) &= \lim_{\eta\to\infty}\left(\frac{\partial p^{(1)}_{\text{in}}}{\partial \eta}(\eta,t)\right)\\
&= \lim_{\eta\to\infty}\left(P_1 p^{(0)}_{\text{in}}(0,t)\frac{1}{\sqrt{D\pi}}\left(1+\frac{1}{2}\exp\left\{-\frac{\eta^2}{4D}\right\}\right) - \frac{P_2}{D}\right)\\
&= \frac{P_1}{\sqrt{D\pi}}p_{\text{out}}(0,t) - \frac{P_2}{D}.
\end{aligned}$$

Here, the second equality employs equation (41), and the final equality uses (47). Multiplying through by the diffusion coefficient D and recalling the macroscopic boundary condition (2),

$$\kappa_1 p_{\text{out}}(0,t) - \kappa_2 = D\frac{\partial p^{(0)}_{\text{out}}}{\partial y}(0,t) = \frac{P_1\sqrt{D}}{\sqrt{\pi}}p_{\text{out}}(0,t) - P_2. \tag{52}$$

Comparing coefficients, we obtain

$$P_1 = \frac{\kappa_1\sqrt{\pi}}{\sqrt{D}} \text{ and } P_2 = \kappa_2. \tag{53}$$

As with the mesoscopic derivation, we have two constraints due to $P_1\sqrt{dt}$ and $2P_2\delta t$ being probabilities. We therefore require:

$$\frac{\kappa_1\sqrt{D\delta t}}{\sqrt{\pi}} < 1 \text{ and } 2\kappa_2\delta t < 1. \tag{54}$$

This concludes the calculation for the parameters of the appropriate microscopic rules. In the next section, we will demonstrate the veracity of the theoretical results that we have obtained in Sections 2 and 3.

4 Results

In this section we will present four examples to demonstrate that the equivalence between the macroscopic boundary condition (2) and its stochastic counterparts is accurate. In all test problems we consider particles diffusing and interacting on a one-dimensional, exponentially growing domain. For a discussion of other types of domain growth, please see the Appendix (Section B). In all examples, we simulate the following PDE and its equivalent stochastic represetations:

$$\frac{\partial u}{\partial t}(x,t) = D\frac{\partial^2 u}{\partial x^2}(x,t) - \rho\frac{\partial (xu(x,t))}{\partial x}, \tag{55}$$

$$D\frac{\partial u}{\partial x}(0,t) = \kappa_1 u(0,t) - \kappa_2, \tag{56}$$

$$u(x,0) = u_0(x), \tag{57}$$

where $x \in (0, L(t))$, $t > 0$ and $L(t) = L_0 e^{\rho t}$ for growth rate $\rho > 0$. Each example has a different combination of κ_1 and κ_2 in order to demonstrate the equivalence across a range of parameter values and conditions. In all of the examples, we choose the diffusion coefficient $D = 0.0025$, the exponential growth rate $\rho = 0.001$ with an initial domain length of $L_0 = 2$, and simulate until a final time of $T = 200$. We simulate each of the stochastic models over a total of 1000 repeats and present density profiles which represent the average of these repeats.

4.1 Example 1: Influx only

In this first example, we set $\kappa_1 = 0$ in order to switch off the partially absorbing boundary, and set $\kappa_2 = 1$ in order to assess how well our equivalences can deal with the constant influx of particles at the left-hand end of the domain with a given rate. We also set the domain to be empty initially, so that $u(x,0) = 0$ for every $x \in (0, L_0)$. The results from this example at time $T = 200$ can be found in Figure 2.

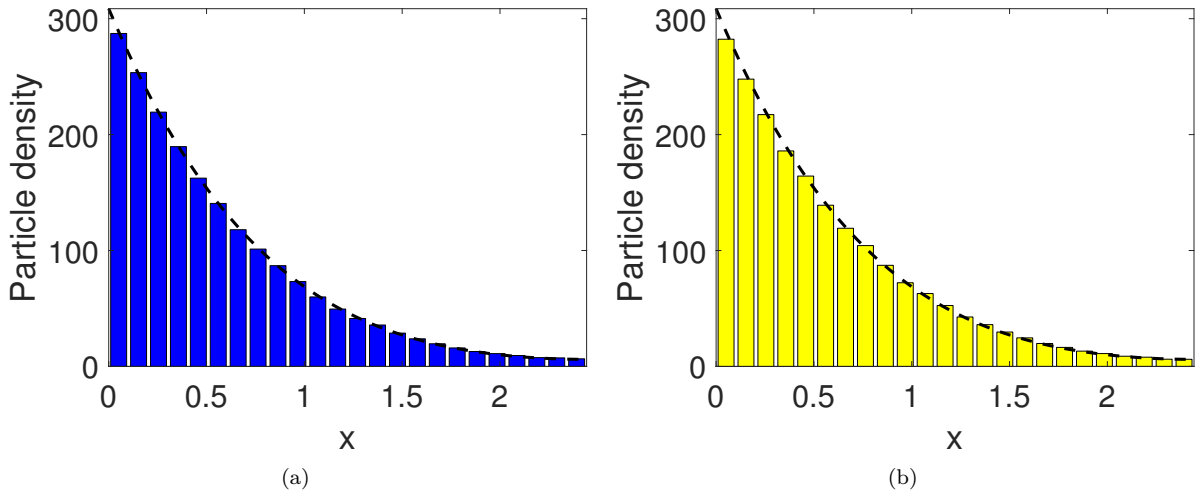


Figure 2. Comparison of the equivalence frameworks for example 1. (a) Comparison between the mesoscale (blue bars) and the macroscopic PDE (black dashed line) and (b) comparison between the microscale (yellow bars) and the PDE. Particles have been binned onto the same mesh as the mesoscale. All parameters are as in the text.

As can be seen, both the mesoscale (Figure 2(a), blue bars) and microscale (Figure 2(b), yellow bars) match the solution of the numerical PDE (black dashed line in each subfigure), indicating good agreement between the three methods.

4.2 Example 2: Partially absorbing only

In the second example, we test the equivalence of the partially absorbing boundary alone. We therefore set $\kappa_2 = 0$ in order to remove the influx, and set $\kappa_1 = 10^{-3}$. We also uniformly place 500 particles throughout the domain initially, equivalent to a PDE initial condition of $u(x, 0) = 250$ for all $x \in (0, L_0)$. A snapshot at time $T = 200$ is presented in Figure 3.

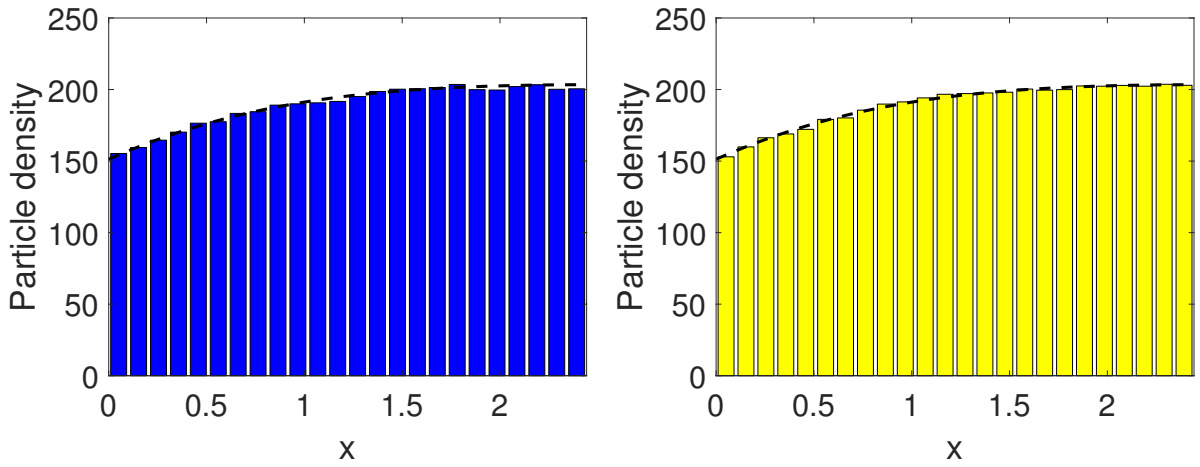


Figure 3. Results for example 2 for (a) the mesoscale and (b) the microscale. All colours are as described in Figure 2, and parameters in the text.

Again, we observe a good agreement between the two stochastic methods and the associated PDE, indicating that the equivalence of the partially absorbing boundary condition is appropriate and accurate.

4.3 Example 3: Partially absorbing reversed

In this example, we use similar parameters to example 2, only we now have a negative κ_1, so that $\kappa_1 = -10^{-3}$ and $\kappa_2 = 0$. These parameter choices capture the physical system in which the creation of a particle occurs with a specified probability when an existing particle hits the boundary. Similar analysis to that undertaken in Sections 2 and 3 suggests that the sign of both Q_1 and P_1 are reversed to accommodate for the negative value of κ_1. Their interpretations are now as follows: $Q_1 h$ is the probability of initialising an additional particle into the first compartment if a particle jumps beyond the boundary; $P_1\sqrt{\delta t}$ is the probability of initialising an additional particle at $x = 0$ given that a particle has crossed the boundary.

The results from this test problem are in Figure 4. As with the previous examples, we once again see a good agreement between the three methods.

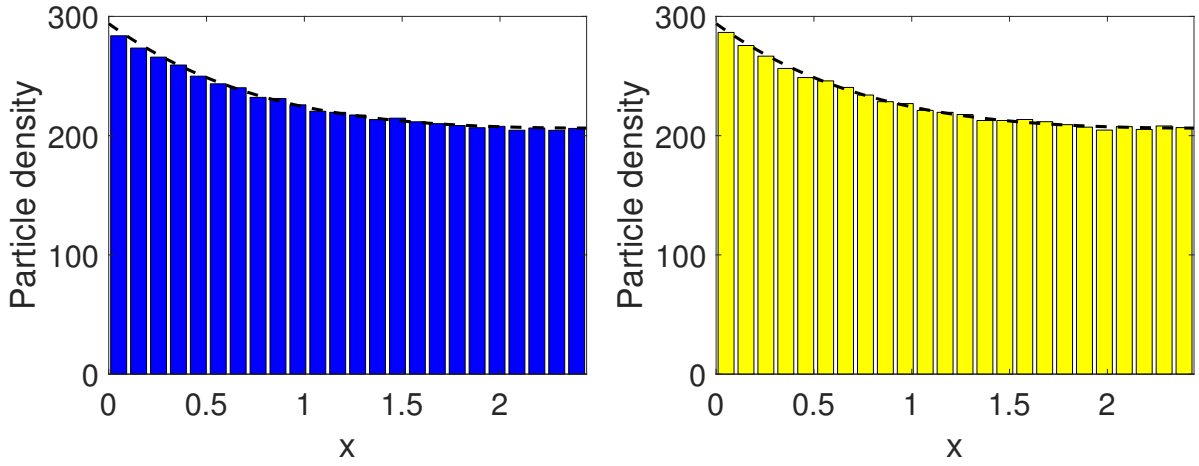


Figure 4. Results for example 3 for (a) the mesoscale and (b) the microscale. All colours are as described in Figure 2, and parameters in the text.

4.4 Example 4: Partially absorbing with influx

In this final example, we apply the two constituent components of the boundary condition simultaneously, and choose $\kappa_1 = 10^{-3}$ and $\kappa_2 = 1$. This will demonstrate that the two components of the boundary condition can be combined accurately. The results from this test problem are found in Figure 5, and show good agreement across all modelling paradigms.

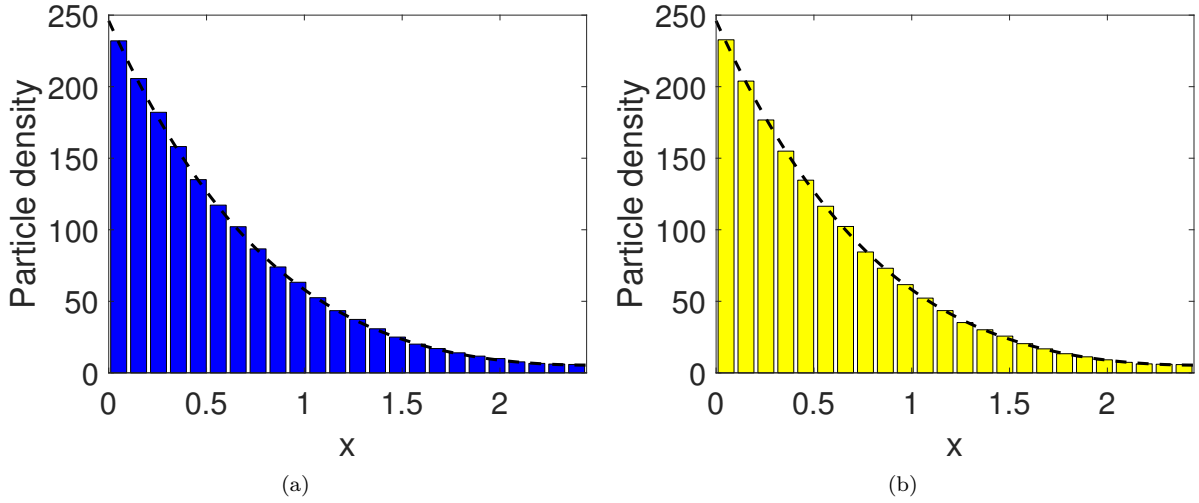


Figure 5. Results for example 4 for (a) the mesoscale and (b) the microscale. All colours are as described in Figure 2, and parameters in the text.

5 Discussion

In this paper, we have derived theoretical stochastic counterparts to the traditional Robin boundary condition for a PDE on a uniformly growing domain. We have demonstrated through a series of simulations, for both the mesoscopic and microscopic representations, that the local rules are able to accurately mimic the appropriate PDE boundary condition. For the mesoscale, we adapted theory produced by ourselves and others (Erban and Chapman, 2007a; Taylor et al., 2015) in order to derive a probability of absorption for a particle provided it has hit the boundary, and a second rate of influx into the first compartment, each of which mimics an aspect of the general Robin boundary condition. For the microscale, we followed an approach pioneered by Singer et al. (2008), and derived absorption and influx probabilities. In all cases we found good agreement between the deterministic and stochastic density profiles which we have demonstrated through a series of numerical examples.

There are many examples from the life sciences which can be modelled as a reaction-diffusion process with a general Robin boundary condition. The formation of a morphogen gradient, for example, is an important concept in developmental biology. The term "morphogen" was initially coined by Alan Turing (Turing, 1952), and describes a particle whose distribution determines the positions of specialised cells and tissue formation. Morphogens are often signalling molecules which originate at some point of a domain, usually a boundary, and diffuse throughout. They are typically modelled using a PDE with a constant influx condition, and with a decay throughout the entire domain. A further example might be the adsorption or particles to boundaries with receptors (Erban and Chapman, 2007b).

With the addition of this equivalence framework for a growing domain, we are able to increase the applicability of stochastic methods to model real-world phenomena. It also facilitates the use and extends the applicability of multi-scale methods such as hybrid methods (Flegg et al., 2012, 2015; Robinson et al., 2014; Smith and Yates, 2018a,b; Yates and Flegg, 2015). These methods utilise different representations of the system in question, for example reaction-diffusion systems, depending on the suitability of that method. PDE methods are fast to simulate but have the drawback that their must be enough particles for a continuum limit to be appropriate. In contrast, stochastic methods are typically slower to simulate but capture individual level detail. In order to couple such methods, an equivalence framework needs to be established, which is precisely what has been achieved in this paper.

We have demonstrated how two stochastic representations for reaction-diffusion systems on growing domains are equivalent to an appropriate PDE with a general Robin boundary condition. These equivalences allow for a wider applicability of stochastic methods for problems in the life sciences where particle-level detail or stochastic fluctuations are important.

Appendices

A Deriving limits for the outer solution

Recall from section 3.2 the definition of $g : \mathbb{R}^3_{\geq 0} \to \mathbb{R}_{\geq 0}$ from equation (48):

$$g(x,y,u) = \frac{1}{\sqrt{4\pi D u}} \exp\left\{ -\frac{\left(y - x - x\frac{\alpha'}{\alpha}u^2\right)^2}{4Du^2} \right\}.$$

We will show that for large values of $y = \eta u$:

$$\lim_{u\to 0} \left(g(x,y,u)\right) \sim \delta(y-x), \tag{58}$$

$$\lim_{u\to 0} \left(g_u(x,y,u)\right) \sim \eta\delta'(y-x). \tag{59}$$

Beginning with equation (58), we note that $g(x, y, u)$ is the pdf of Gaussian random variable with mean $x + x\frac{\alpha'}{\alpha}u^2$ and variance $2Du^2$, so $Y \sim N\left(x + x\frac{\alpha'}{\alpha}u^2, 2Du^2\right)$. Define a new random variable $Z \sim N(0, 2Du^2)$, and further, let $f_Z(z, u)$ be the pdf of Z. In the distributional sense, we have

$$\lim_{u\to 0}\left(f_Z(z, u)\right) \sim \delta(z).$$

This means that for smooth functions $\varphi \in C_C^\infty$ (C_C^∞ is the set of infinitey differentiable functions with compact support on $\mathbb{R}$), that

$$\varphi(0) = \lim_{u\to 0}\left(\int_{\mathbb{R}} f_Z(z; u)\varphi(z)\ \mathrm{d}z\right), \tag{60}$$

where we have made the dependence of f_Z on u explicit. Since the difference between Y and Z is a translation (so that the mean value moves from $z = 0$ to $y = x + x\frac{\alpha'}{\alpha}u^2 \approx x$ for small u), we obtain the first limit:

$$\lim_{u\to 0}\left(g(x, y, u)\right) \sim \delta(y - x). \tag{61}$$

For equation (59), we firstly define a new variable ζ, which is a combination of x, y and u:

$$\zeta(x, y, u) = \frac{y - x - x\frac{\alpha'}{\alpha}u^2}{\sqrt{2D}u}, \tag{62}$$

which is defined so that $g(x, y, u) = \frac{1}{\sqrt{4\pi D}u}\exp\left\{-\frac{\zeta(x,y,u)^2}{2}\right\}$. Then, we can write the derivative of g with respect to u as (writing $\zeta(x, y, u) = \zeta(u)$ for brevity):

$$\frac{\partial g}{\partial u}(x, y, u) = \left[\frac{1}{u}\left(y - x + x\frac{\alpha'}{\alpha}u^2\right) - \frac{1}{u}\right]\frac{1}{\sqrt{2D}u}g(x, y, u) \tag{63}$$

As with the inner solution, we let $y = \eta u$ and $x = \chi u$, where we recall that u has replaced $\sqrt{\delta t}$. Substituting these into equation (63), together with $\zeta_y(x, y, u) = 1/(\sqrt{2D}u)$, we obtain:

$$\frac{\partial g}{\partial u}(x, y, u) = \left[\left(\eta - \chi + \chi\frac{\alpha'}{\alpha}u^2\right) - \frac{1}{u}\right]g(x, y, u)\zeta_y(x, y, u).$$

We will eventually be taking limits as $\eta \to \infty$, so we assume that $0 < \chi \ll \eta$, and so, asymptotically:

$$\frac{\partial g}{\partial u}(x, y, u) \sim \eta\zeta_y(u)g(x, y, u).$$

Finally, by direct computation we note that, up to a constant:

$$\zeta_y(u)g(x, y, u) \approx \frac{\partial g}{\partial y}(x, y, u),$$

which it can be shown tends towards $\delta'(y - x)$ as $u \to 0$ by a similar argument to that in equation (60). Therefore:

$$\lim_{u\to 0}\left(g_u(x, y, u)\right) \sim \eta\delta'(y - x), \tag{64}$$

which holds for large values of η.

B Examples of growth

In this section, we briefly describe several types of uniform growth, and how the various functions and parameters behave under each of these types of growth. We note that there are many different types of domain growth, however in this section, we describe exponential, linear and logistic. All parameters and functions under these growth types are summarised in Table 1.

Type of growth	$\alpha(t)$	$r(k)$
Exponential	$e^{\rho t}$	ρ
Linear	$1+\rho t$	$\frac{\rho}{hk}$
Logistic	$\frac{e^{\rho t}}{1+\omega^{-1}(e^{\rho t}-1)}$	$\rho\left(1-\frac{k}{\gamma k_0}\right)$

Table 1. Different types of domain growth and how these are matched to the functions in the main text. The second column is the stretching function, which appears in the PDE (1) and the SDE (9). The third column is the rate at which new compartments are added into the mesoscale. These equivalences are shown in (Baker et al., 2010) for a slightly different domain growth mechanism. All parameters are described in the text.

Exponential growth

We firstly describe exponential growth. This is the example employed in the results section (Section 4) because it is the simplest to employ. In this case, the stretching function $\alpha(t)$ is defined as

$$\alpha_{\text{exp}}(t) = e^{\rho t},$$

where ρ is the exponential growth rate. This means that the PDE (1) is the same as the one that appears in the results (equation (55)). For the mesoscale, we need to match this to the rate at which a new compartment (for each compartment) is added to the domain. It can be shown (Baker et al., 2010) that this equivalence is

$$r_{\text{exp}}(k) = \rho,$$

so that a new compartment is added with rate $kr_{\text{exp}}(k) = \rho k$.

Linear growth

For linear growth, we have a stretching function defined by

$$\alpha_{\text{lin}}(t) = 1 + \rho t,$$

where ρ is again the rate of growth. This choice yields the following PDE

$$\frac{\partial u}{\partial t}(x,t) = D\frac{\partial^2 u}{\partial x^2}(x,t) - \frac{\rho}{1+\rho t}\frac{\partial (xu(x,t))}{\partial x}.$$

We match this to the rate per compartment of adding new compartments by defining

$$r_{\text{lin}}(k) = \frac{\rho}{hk},$$

so that the appropriate rate that a new compartment is added across the whole domain is $kr_{\text{lin}}(k) = \rho/h$ which remains constant.

Logistic growth

Logistic growth is defined so that the derivative of the stretch function satisfies the logistic equation:

$$\frac{\mathrm{d}\alpha_{\log}}{\mathrm{d}t}(t) = \rho\alpha_{\log}(t)\left(1 - \frac{\alpha_{\log}(t)}{\gamma}\right), \tag{65}$$

where ρ is the growth rate and ω is the carrying capacity which is the maximum scale factor by which the domain is allowed to grow. Using $\alpha_{\log}(0) = 1$ we solve the equation (65) to find:

$$\alpha_{\log}(t) = \frac{e^{\rho t}}{1 + \omega^{-1}(e^{\rho t} - 1)},$$

and the corresponding PDE:

$$\frac{\partial u}{\partial t}(x,t) = D\frac{\partial^2 u}{\partial x^2}(x,t) - \rho\left(1 - \frac{\alpha_{\log}(t)}{\gamma}\right)\frac{\partial(xu(x,t))}{\partial x}.$$

Matching this to the mesoscale requires us to define

$$r_{\log}(k) = \rho\left(1 - \frac{k}{\gamma k_0}\right),$$

where k_0 is the initial number of compartments. Note that while this is defined for every $k \in \mathbb{N}$, since we are interested in domain growth, we make the assumption that $k < \gamma k_0$. The rate of adding a new compartment therefore mimics the right hand side of the logistic equation:

$$kr_{\log}(k) = \rho k\left(1 - \frac{k}{\omega k_0}\right).$$

References

D.F. Anderson. A modified next reaction method for simulating chemical systems with time dependent propensities and delays. *J. Chem. Phys.*, 127(21):214107, 2007.

R.E. Baker, C.A. Yates, and R. Erban. From microscopic to macroscopic descriptions of cell migration on growing domains. *Bull. Math. Biol.*, 72(3):719–762, 2010.

E.J. Crampin, E.A. Gaffney, and P.K. Maini. Reaction and diffusion on growing domains: scenarios for robust pattern formation. *Bull. Math. Biol.*, 61(6):1093–1120, 1999.

D.C. Deeming and M.W.J. Ferguson. Morphometric analysis of embryonic development in alligator mississippiensis, crocodylus johnstoni and crocodylus porosus. *J. Zool.*, 221(3):419–439, 1990.

R. Erban and S.J. Chapman. Reactive boundary conditions for stochastic simulations of reaction–diffusion processes. *Phys. Biol.*, 4(1):16–28, 2007a.

Radek Erban and S Jonathan Chapman. On chemisorption of polymers to solid surfaces. *Journal of Statistical Physics*, 127(6):1255–1277, 2007b.

M.B. Flegg, S.J. Chapman, and R. Erban. The two-regime method for optimizing stochastic reaction–diffusion simulations. *J. Roy. Soc. Interface.*, 9(70):859–868, 2012.

M.B. Flegg, S. Hellander, and R. Erban. Convergence of methods for coupling of microscopic and mesoscopic reaction-diffusion simulations. *J. Comput. Phys.*, 289(C):1–17, May 2015. ISSN 0021-9991.

P. Gerlee. The model muddle: in search of tumor growth laws. *Cancer Res.*, 73(8):2407–2411, 2013.

M.A. Gibson and J. Bruck. Efficient exact stochastic simulation of chemical systems with many species and many channels. *J. Phys. Chem. A.*, 104(9):1876–1889, 2000.

D.T. Gillespie. Exact stochastic simulation of coupled chemical reactions. *J. Phys. Chem.*, 81(25): 2340–2361, 1977.

D.T. Gillespie. Approximate accelerated stochastic simulation of chemically reacting systems. *J. Chem. Phys.*, 115(4):1716–1733, 2001.

D.W. Hahn and M.N. Özisik. *Heat conduction*. John Wiley & Sons, 2012.

P.E. Kloeden and E. Platen. *Numerical solution of stochastic differential equations*, volume 23. Springer Science & Business Media, 2013.

B. Korsgaard and F.Ø. Andersen. Embryonic nutrition, growth and energetics inzoarces viviparus l. as indication of a maternal-fetal trophic relationship. *J. Comp. Physiol. B*, 155(4):437–444, 1985.

P.M. Kulesa, G.C. Cruywagen, S.R. Lubkin, P.K. Main, J. Sneyd, M.W.J. Ferguson, and J.D. Murray. On a model mechanism for the spatial patterning of teeth primordia in the alligator. *J. Theor. Biol.*, 180(4):287–296, 1996.

P.M. Kulesa, C.M. Bailey, J.C. Kasemeier-Kulesa, and R. McLennan. Cranial neural crest migration: new rules for an old road. *Dev. Biol.*, 344(2):543–554, 2010.

A. Leshem, A. Ar, and R.A. Ackerman. Growth, water, and energy metabolism of the soft-shelled turtle (trionyx triunguis) embryo: effects of temperature. *Physiol. Zool.*, 64(2):568–594, 1991.

R. McLennan, L. Dyson, K.W. Prather, J.A. Morrison, R.E. Baker, P.K. Maini, and Kulesa P.M. Multiscale mechanisms of cell migration during development: theory and experiment. *Development*, 139 (16):2935–2944, 2012.

R.L. Mort, R.J.H. Ross, K.J. Hainey, O.J. Harrison, M.A. Keighren, G. Landini, R.E. Baker, K.J. Painter, I.J. Jackson, and C.A. Yates. Reconciling diverse mammalian pigmentation patterns with a fundamental mathematical model. *Nat. Commun.*, 7:10288, 2016.

J.D. Murray, D.C. Deeming, and M.W.J. Ferguson. Size-dependent pigmentation-pattern formation in embryos of alligator mississippiensis: time of initiation of pattern generation mechanism. *Proc. Royal Soc. B*, 239(1296):279–293, 1990.

H. Risken. Fokker-planck equation. In *The Fokker-Planck Equation*, pages 63–95. Springer, 1996.

M. Robinson, M.B. Flegg, and R. Erban. Adaptive two-regime method: application to front propagation. *J. Chem. Phys.*, 140(12):124109, 2014.

A. Singer and Z. Schuss. Brownian simulations and unidirectional flux in diffusion. *Phys. Rev. E*, 71(2): 026115, 2005.

A. Singer, Z. Schuss, A. Osipov, and D. Holcman. Partially reflected diffusion. *SIAM J. Appl. Math.*, 68 (3):844–868, 2008.

C.A. Smith and C.A. Yates. The auxiliary region method: A hybrid method for coupling a PDE to Brownian-based dynamics for reaction-diffusion systems. *R. Soc. Open Sci.*, 5(8), 2018a.

C.A. Smith and C.A. Yates. Spatially extended hybrid methods: a review. *J. Roy. Soc. Interface*, 15 (139), 2018b.

C.A. Smith, C Mailler, and C.A Yates. Unbiased on-lattice domain growth. *Phys. Rev. E*, 100(6), dec 2019. doi: 10.1103/physreve.100.063307.

P.R. Taylor, R.E. Baker, and C.A. Yates. Deriving appropriate boundary conditions, and accelerating position-jump simulations, of diffusion using non-local jumping. *Phys. Biol.*, 12(1):016006, 2015.

A.M. Turing. The chemical basis of morphogenesis. *Phil. Trans. R. Soc. B.*, 237(641):37–72, 1952.

T.E. Woolley, R.E. Baker, E.A. Gaffney, and P.K. Maini. Stochastic reaction and diffusion on growing domains: understanding the breakdown of robust pattern formation. *Phys. Rev. E*, 84(4):046216, 2011.

C.A. Yates and M.B. Flegg. The pseudo-compartment method for coupling partial differential equation and compartment-based models of diffusion. *J. Roy. Soc. Interface*, 12(106):20150141, 2015.

5.2 Conclusions

We have developed an equivalence framework for a generic Robin boundary condition on a uniformly growing domain, in which we match local rules for the mesoscopic and microscopic representations to the macroscopic boundary condition defined via a PDE. We have demonstrated that the equivalence is accurate through a series of test problems which match the PDE to each of the stochastic methods. This is the first time such a calculation and equivalence has been calculated for systems on growing domains.

The results in this paper will allow for a wider range of biological examples to be modelled using the two stochastic representations (mesoscale and microscale). Such examples may include the formation of a morphogen gradient on a growing domain (Yates, 2014), which would require the addition of particles at the boundary, together with degradation everywhere across the domain.

The next chapter contains the paper which has been the main objective of the thesis and which ties together all of its strands. We develop three spatially extended hybrid methods on uniformly growing domains. The work presented in this chapter will be utilised when we test each of these hybrid methods. One of the key tests for hybrid methods is whether it is able to correctly resolve a system which has a flux over the interface. This boundary condition equivalence developed in this chapter allows us to generate an example which guarantees such a flux over the interface in order to ensure that the hybrid methods are accurate in this case. This paper will further be utilised in the morphogen gradient example of Chapter 6, which demonstrates that our hybrid methods on growing domains are capable of incorporating reactions.

Chapter 6

Incorporating domain growth into hybrid methods for reaction-diffusion systems

This chapter contains a paper authored by Kit Yates and myself, which has been submitted to the *Journal of the Royal Society: Interface*, together with the accompanying supplementary material. This paper details the development of three spatially extended hybrid methods on uniformly growing domains. There are three methods created, each of which builds upon a method from the literature, namely the pseudo-compartment method (PCM) (Yates and Flegg, 2015), the ghost cell method (GCM) (Flegg et al., 2015) and the auxiliary region method (ARM) (Smith and Yates, 2018). Throughout the paper, we draw on each of the previous chapters in order to formulate and test these methods. We extend the ARM, which is developed in Chapter 3, the stretching method (Chapter 4) is employed in both the extensions of the PCM and GCM, while the equivalence framework for the Robin boundary condition detailed in Chapter 5 is utilised in order to test each of the growing hybrid methods. Each of the test problems compare the three hybrid methods to the mean-field PDE obtained by forming the RDME for the underlying process (diffusive, growth and reaction systems) and taking the diffusive limit in order to obtain the PDE that we compare to.

As with the previous chapters, we do not consider reactions of a higher order than one throughout this chapter in order to attribute the discrepancy between the developed hybrid methods and their associated mean-field PDEs to the hybrid methods themselves. When considering higher order reactions there are several problems that need

to be considered. At the mesoscale, if second-order reactions are present in the system, as the size of compartments decreases, second-order reactions will eventually be lost due to the fact that, traditionally, particles need to be in the same compartment in order to react. Isaacson (2013) develops a convergent RDME (CRDME), which allows particles to react further afield in order to prevent the effective depletion of second-order reactions, meaning that the RDME is convergent in the small compartment limit. At the microscale, we require a second-order method such as the λ-ρ method (Erban and Chapman, 2009). The main problem with such methods is that, close to the interface, the reaction radius for any particle may intersect with the other subdomain (be it the macro or mesoscale). A potential avenue for fixing this issue is given in Section 3.6 of the paper in Chapter 3, however, it remains open work as to whether this is appropriate on a growing domain. We also do not consider volume exclusion in the methods that we have developed. However, the ideas behind each hybrid algorithm are independent of the individual methods that are employed. For example, provided that the number of particles within a region can be calculated for a PDE update algorithm, the finite-difference scheme that we employ can be replaced. The same would hold for the meso and microscales too.

This paper represents the first time that spatially extended hybrid methods have been implemented on uniformly growing domains, however there have been other papers that do extend other types of hybrid methods of different types onto growing domains, one of which is described in the discussion following the paper (McLennan et al., 2012). The development of such methods is important because domain growth is an inherent biological mechanism for many processes in the life sciences. The paper focusses on exponential growth as an example, which has been shown to be the growth type for the growth of alligator jaws (Deeming and Ferguson, 1990; Kulesa et al., 1996; Murray et al., 1990). Other growth types could be linear growth, which is observed in the early development of fish (Korsgaard and Andersen, 1985) and logistic growth is exhibited during the embryonic growth of turtles (Leshem et al., 1991).

6.1 Overview of the paper

The paper in this chapter is set out as follows. The first section contains an introduction to spatially extended hybrid methods and the three hybrid methods that are to be developed, drawing on the review paper from Chapter 2 and the development of the auxiliary region method from Chapter 3. In Section 2, we briefly explain the three

individual modelling paradigms and how they can be considered equivalent. In Sections 3 to 5, we develop the growing versions of the PCM, GCM and ARM respectively, including algorithms for their implementation. In Section 6, we use three test problems in order to assess each of the developed methods. Two of these test problems employ the methods from Chapter 5 in order to ensure that all comparisons are valid from an equivalence point of view. Finally, in Section 7, we discuss the findings of the paper, and place them into a wider mathematical and biological context.

Appendix 6B: Statement of Authorship

This declaration concerns the article entitled:			
Incorporating domain growth into hybrid methods for reaction-diffusion systems			
Publication status (tick one) **Draft manuscript** ☐ **Submitted** ● **In review** ☐ **Accepted** ☐ **Published** ☐			
Publication details (reference)	Journal – Journal of the Royal Society: Interface Authors – Cameron A. Smith, Christian A. Yates		
Copyright status (tick the appropriate statement) I hold the copyright for this material ● Copyright is retained by the publisher, but I have been given permission to replicate the material here ☐			
Candidate's contribution to the paper (provide details, and also indicate as a percentage)	All calculations have been performed by the author of this thesis (100%). All numerical computations and simulations have been completed by the author of this thesis (100%). All authors contributed equally to the presentation of the content (50%).		
Statement from Candidate	This paper reports on original research I conducted during the period of my Higher Degree by Research candidature.		
Signed		**Date**	04/06/2021

Incorporating domain growth into hybrid methods for reaction-diffusion systems

Cameron A. Smith[1,*], Christian A. Yates[1]
[1]**Centre for Mathematical Biology, Department of Mathematical Sciences, University of Bath, Claverton Down, Bath, BA2 7AY, United Kingdom**

Key words: Reaction–diffusion, domain growth, hybrid methods.

Abstract

Reaction–diffusion mechanism are a robust paradigm that can be used to represent many biological and physical phenomena over multiple spatial scales. Applications include intracellular dynamics, the migration of cells and the patterns formed by vegetation in semi-arid landscapes. Moreover, domain growth is an important process for embryonic growth and wound healing. There are many numerical modelling frameworks capable of simulating such systems on growing domains, however each of these may be well suited to different spatial scales and particle numbers. Recently, spatially extended hybrid methods on static domains have been produced in order to bridge the gap between these different modelling paradigms in order to represent multiscale phenomena. However, such methods have not been developed with domain growth in mind. In this paper, we develop three hybrid methods on growing domains, extending three of the prominent static domain hybrid methods. We also provide detailed algorithms to allow others to employ them. We demonstrate that the methods are able to accurately model three representative reaction-diffusion systems accurately and without bias.

1 Introduction

The reaction–diffusion paradigm can be employed to model a range of biological and physical scenarios over multiple length scales, from representing vegetation patterns in semi-arid landscapes (Sherratt, 2005) and the study of epidemics (Volpert and Petrovskii, 2009), to intracellular dynamics (Andasari et al., 2012; Khan et al., 2011; ZhuGe et al., 2000). These systems couple the random movement of particles (which when considered at a continuum level manifests as the movement of particle density down the concentration gradient) and the interaction of particles with each other and potentially with the domain boundaries.

Domain growth is a process which underpins many biological processes, and it is therefore important that we have accurate and efficient modelling methods to represent it. Examples span many biological applications, including the growth and shrinkage of tissue (Wolpert et al., 2015; Yates, 2014) and neural crest cell migration (Kulesa et al., 2010; McLennan et al., 2012; Mort et al., 2016). Domain growth has also been shown to play an important role in theoretical studies of pattern formation in reaction-diffusion systems (Crampin et al., 1999; Taylor et al., 2015; Woolley et al., 2011).

Reaction–diffusion systems can be modelled in several different ways, each of which has different suitabilities depending on the scale of the system being modelled. The coarsest of the three methods that we focus on is the macroscale, which uses partial differential equations (PDEs) in order to represent the system. PDEs model how the continuum *density* of particles evolve in time, and are only suitable if the number of particles is high enough to consider a continuum limit. The reaction–diffusion PDE on a growing domain consists of four components (see equation (1)): a first order differential of density with respect to time which describes the change of concentration in time; a second-order differential of density with respect to space which represents diffusion; a first-order differential of density with respect to space representing domain growth, and finally a term which represents reactions, if any are present. The macroscale is generally quick to implement using a number of established techniques (for example, see Brenner and Carstensen (2004); Eymard et al. (2000); Morton and Mayers (2005); Smith (1985)), and there are often analytical approaches that can be employed to investigate such systems. However, if particle numbers are too low, the assumption that the continuum limit holds may break down, and

stochastic fluctuations may be found to play a more pivotal role. Moreover, the deterministic mean-field PDE may not fully agree with its stochastic counterparts described below. Attempts to derive a deterministic equivalent to a non-linear stochastic model may result in an infinite hierarchy of interrelated equations. This results in the need for moment closure, which necessarily leads to the loss of some of the information encapsulated in the higher-order moments.

The second modelling paradigm that we consider is the mesoscale, where we split the spatial domain into a series of compartments, within which particles reside. These particles are able to jump between neighbouring compartments, and are able to interact/react with others within their own compartment. These events are given exponential waiting times whose rates dictate the evolution of the system. There are a large number of algorithms to simulate such systems, both exact (Anderson, 2007; Gibson and Bruck, 2000; Gillespie, 1977) and approximate (Auger et al., 2006; Gillespie, 2001). This middle scale is generally slower than the macroscale, but provides more fine-grained accuracy when particle numbers are lower. In order to incorporate domain growth at least two methods that have been proposed. The first is a local method proposed by Baker et al. (2010), which chooses a compartment uniformly at random to instantaneously double in size and then divide. The particles in the parent compartment are distributed into the two daughter compartments according to some symmetric probability distribution. This method causes a build-up of particles at the boundaries when growth occurs on a faster time-scale than diffusion (Smith et al., 2019). The second method (and the method we will employ in this work), is a global method introduced by Smith et al. (2019). Compartments grow uniformly until a growth event is due to occur. At this point, the boundaries between compartments are redrawn to include one extra compartment, and the particles are redistributed appropriately. This method works well in both high and low diffusion regimes.

Finally, at the microscale, we investigate Brownian-based dynamics. At this, the smallest scale that we employ, individual particles are tracked and updated in continuous space. There are several techniques that can be employed in order to simulate a system at this scale, including the time-driven mechanisms of Brownian motion for purely diffusive processes and Smoluchowski dynamics (Smoluchowski, 1917) when reactions are involved, or the event-driven Green's function reaction dynamics (GFRD) (van Zon and ten Wolde, 2005). Under time-driven algorithms, particles diffuse and are diluted according to an appropriate stochastic differential equation (SDE). As well as each particle's location, if the system requires second- or higher-order reactions, we also need to calculate the pairwise distances between all particles at every time-step, which means that while this modelling paradigm is the most accurate, it is also the slowest. Added to this, if the system in question is diffusion limited, a very small time-step is required to accurately resolve the dynamics. More efficient time-stepping is employed by the event-driven GFRD. This sets a maximum time-step and the solution to the Smoluchowski equation in order to combine diffusion and interactions, whilst accounting for the additional error which is introduced in doing so.

There is another, even finer, scale of spatially resolved model known as molecular dynamics, which we do not consider in this paper. We direct the interested reader to (Dürr et al., 1981; Holley, 1971) for more information.

Many biological problems of interest are genuinely multiscale (Black and McKane, 2012; Gillespie et al., 2013; Markevich et al., 2004; Robinson et al., 2014). Consequently we require methods that are able to resolve the dynamics at the appropriate scale. Spatially extended hybrid methods (Smith and Yates, 2018b) are one such class of techniques that are able to do this. These methods employ two or more reaction-diffusion modelling paradigms (described above) to represent the dynamics in different areas of the domain in the most appropriate way. In regions with low particle numbers, for example, one of the finer-grained stochastic methods might be employed at the cost of reduced simulation efficiency. However, in regions of high particle numbers — enough to consider a continuum limit — the more computationally efficient PDE may be used. There are many examples of spatially extended hybrid methods (Alexander et al., 2002; Erban, 2014, 2016; Flegg et al., 2012, 2015; Franz et al., 2013; Moro, 2004; Robinson et al., 2014; Smith and Yates, 2018a; Spill et al., 2015; Yates and Flegg, 2015), all of which focus on a static

(non-growing) domain. This paper extends three of these methods onto uniformly growing domains. The pseudo-compartment method (PCM) (Yates and Flegg, 2015) is a macroscopic-to-mesoscopic method which uses an interface to divide the domain into two subdomains. Particles are able to jump between the two subdomains via a 'pseudo-compartment' adjacent to the interface within the PDE subdomain. The ghost cell method (GCM) proposed by Flegg et al. (2015) is a mesoscopic-to-microscopic method that makes similar use of an extra compartment, coined the ghost cell, adjacent to the interface in the microscopic subdomain in order to couple compartment-based and Brownian dynamics. Finally, the auxiliary region method (ARM) (Smith and Yates, 2018a) is a macroscopic-to-microscopic method which, similar to the PCM and GCM, which employs a mesoscale auxiliary region at the interface in order to allow particles to jump between the two subdomians.

The rest of this paper will be set out as follows. In Section 2 we briefly explain how the macroscopic, mesoscopic and microscopic can be considered equivalent to each other. In Sections 3–5 we introduce the PCM, GCM and ARM, explain the key differences between the algorithms for a static and growing domain, and present the algorithms in full. We present representative results for multiple test problems in Section 6, and discuss our findings in Section 7.

2 Equivalence Framework

In this section we present each of the three modelling paradigms that are the constituent parts of our three hybrid methods. In Section 2.1 we introduce the PDE approach. We present the mesoscopic approach in Section 2.2 and briefly demonstrate how we can consider it to be equivalent to the PDE in the appropriate limit. Finally, in Section 2.3 we do the same for the individual, particle-based dynamics. All numerical algorithms can be found in the Supplementary Material.

2.1 Macroscopic modelling

Firstly we consider the macroscale PDE representation. Consider a population with density $u(x,t)$ undergoing diffusion at a position x on an exponentially growing one-dimensional domain $(0, L(t)) \subseteq \mathbb{R}$ at time $t > 0$. We consider just a single dimension and exponential growth (with rate ρ) for ease of description here, but the methods outlined can be extended to higher dimensions, and will work for any form of uniform growth. Under these assumptions, the length of the domain is $L(t) = L_0 \exp\{\rho t\}$, where L_0 is the initial length of the domain, and the concentration evolves according to the following PDE:

$$\frac{\partial u}{\partial t}(x,t) = D\frac{\partial^2 u}{\partial x^2}(x,t) - \rho\frac{\partial (xu(x,t))}{\partial x}, \tag{1}$$

which holds for $x \in (0, L(t))$ and for $t > 0$. This description of domain growth is known as the Eulerian representation in Eulerian coordinates, x and t. The first term on the right-hand side represents the spread of particles due to diffusion (with Fickian diffusion coefficient D) and the second term is the dilution and spread of concentration caused by the stretching of the spatial domain. We will employ this PDE in order to demonstrate the equivalence of the two finer-scale methods to this first one.

In order to simulate this PDE, we need to switch from the Eulerian coordinates above, in which the domain grows in time, to Lagrangian coordinates, in which the domain remains static in time. We do this through the following change of coordinates:

$$x = X \exp\{\rho t\}, \tag{2}$$

$$t = \tau. \tag{3}$$

Note now that the length of the domain in the Lagrangian spatial variable X ranges between 0 and the fixed value L_0. Further, the Lagrangian and Eulerian temporal variables, τ and t respectively, coincide

here. There are times when it is useful to rescale time, such as when finding analytical solutions to the diffusion equation on uniformly growing domains (Simpson, 2015). For more details on the Lagrangian PDE and how we solve it numerically, please see Section S.1.2 of the supplementary material (SM).

2.2 Mesoscopic modelling

In this section, we describe the mesoscale representation, which is the "middle" scale that we will consider. In order to model at this level, we will divide the spatial domain $(0, L(t))$ into a number of compartments labelled $C_i(t)$ for $i \in \{1, ..., K(t)\}$, where $K(t)$ is the (time-dependent) number of compartments at time t. Particles lie within these compartments and are able to jump between neighbouring compartments mimicking diffusion, and can interact with one another within the same compartment through R reaction channels. We will define the state of the system at time t to be $\boldsymbol{N}(t)$, where $N_i(t)$ is the number of particles at time t in compartment i. Throughout this paper, we will implement the modified next reaction method (Anderson, 2007) in order to advance the system forwards in time. This method is used for explicit time-varying propensity functions (the propensity function is a proxy for the rate of that event occurring), which are important due to the domain growth. For an explanation of the method and the algorithm, see the Supplementary Material, Section S.2.1.

To extend the domain, we utilise the stretching method of Smith et al. (2019). As the domain grows through a series of discrete fixed size extensions, the compartments grow in length uniformly. Once a stretch event has been determined to occur, the compartment boundaries are redrawn, making each compartment smaller and making way for a new compartment. The particles are then appropriately redistributed assuming a uniform distribution of particles across each compartment. This algorithm can be included in the main mesoscale algorithm in two ways. Firstly, propensity functions for the splitting events (events where a new compartment is added) can be added to the list of propensity functions, making this stochastic in time and space. Alternatively (and the method we utilise here), we calculate the deterministic time at which new compartments should be added to effect a particular domain growth pattern, and add them at this time. In this way, the growth events are still stochastic in space, but are now time-deterministic.

It can be shown, in the limit of small compartment size and fast inter-compartment jumping rate, that the average behaviour of this mesoscopic model can be described by the Eulerian PDE (1). We briefly present the main steps of the calculation in the Supplementary Material (Section S.2.3). For a more complete calculation please see (Smith et al., 2019) for the mean equations and Section S.2.3 in the Supplementary Material for the small compartment limit.

2.3 Microscopic modelling

Within this section, we introduce the finest scale modelling paradigm that we will employ for the hybrid methods presented in this paper. The microscale tracks the individual locations of particles which diffuse and are repositioned (due to domain growth) according to a stochastic differential equation (SDE) as well as interacting depending on proximity to one another. Particles diffuse through a Brownian motion, and the growth is implemented using a deterministic drift term. In practice, this means that each particles is "pulled along" as the domain stretches.

Let $X_t \in (0, L(t))$ be the position of a particle at time t. Then this evolves according to:

$$\mathrm{d}X_t = \rho X_t \, \mathrm{d}t + \sqrt{2D} \, \mathrm{d}W_t. \tag{4}$$

Here, the term on the left hand side denotes the change in position, the first term on the right hand side is the drift term representing the repositioning of the particles caused by the stretching of the domain, and the second term on the right hand side is the diffusion. Note also that $\mathrm{d}W_t$ is a standard Weiner process.

In order to show that the density of particles evolving according to this SDE is described by the Eulerian PDE (1), we look to the Fokker-Planck equation (FPE) (otherwise known as the Kolmogorov forward equation (KFE)) (Risken, 1996). As with the mesoscopic case, we briefly outline the derivation in the SM (Section S.3.1), and refer the interested reader to the aforementioned reference.

3 The pseudo-compartment method

The first hybrid method we will adapt to incorporate uniform domain growth is the macroscopic-to-mesoscopic pseudo-compartment method (PCM) (Yates and Flegg, 2015). At time t we decompose the domain as follows. The PDE subdomain occupies the region $\Omega_{\text{P}}(t) = (0, I(t))$, where $I(t)$ is the location of the interface at time t, and the mesoscopic subdomain is $\Omega_{\text{C}}(t) = (I(t), L(t))$, where $L(t)$ is the total length of the domain. The values of $I(t)$ and $L(t)$ will be calculated deterministically from the initial position of these boundaries, and the growth process, via the following pair of equations

$$I(t) = I(0)\exp\{\rho t\},$$
$$L(t) = L(0)\exp\{\rho t\}.$$

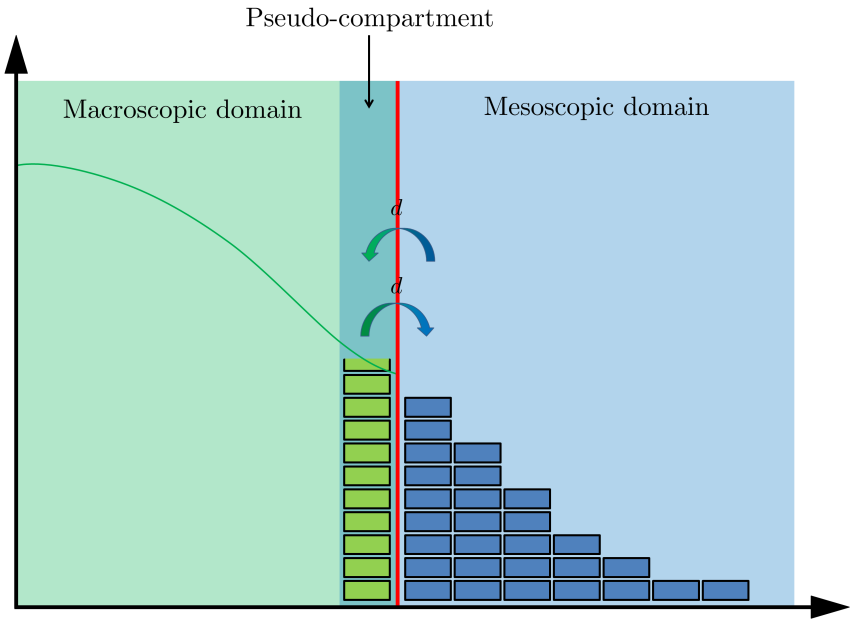


Figure 1. A schematic of the static PCM (Yates and Flegg, 2015). The green line denotes the density of particles in the macroscopic domain, while the blue rectangles represent particles within each compartment. The red line is the interface, and the green rectangles represent the number of "pseudo-particles", obtained by direct integration of the PDE solution over the pseudo-compartment. The arrows crossing the interface denote the movement of the pseudo-particles between the two subdomains.

In order to couple the macroscale and mesoscale, we allow particles to jump between the first mesoscopic compartment (next to the interface (vertical red line) in Figure 1) and the so-called "pseudo-

compartment", which is a region of space the width of one compartment next to the interface on the PDE side of the domain. Mass may only cross the interface through this jumping mechanism and not though continuum PDE mass flowing over the boundary. As such, a zero-flux boundary condition is imposed on the PDE at the interface.

Suppose that the number of particles in compartment $C_1(t)$ at time t is $n_1(t)$, and that it is of width $h_c(t)$. Then the region of space occupied by the pseudo-compartment of length $h_{PC}(t)$ is denoted $C_{-1}(t)$ and contains $n_{-1}(t)$ particles, where

$$n_{-1}(t) = \int_{C_{-1}(t)} u(x,t)\ dx. \tag{5}$$

In order to incorporate jumps into and out of the pseudo-compartment, we introduce two events into the list of mesoscopic events, one for a jump into the pseudo-compartment from compartment 1, and one for a jump in the other direction. Since, on the growing domain these compartments are potentially of different widths, the appropriate jumping rates would be $D/(h_c(t)^2)$ and $D/(h_{PC}(t)h_c(t))$ respectively. The second of these takes into account the differing compartment sizes. In the case that $h_c(t)$ and $h_{PC}(t)$ are the same, this collapses to the usual diffusive jump rate.

The reasons that the length of a "regular" compartment $h_c(t)$ is typically different from the length of the pseudo-compartment $h_{PC}(t)$ are subtle and due to the differences between the Lagrangian (static) and Eulerian (growing) coordinates. The PDE is being solved using a fixed mesh-width in Lagrangian coordinates, and as such, the mesh grows when considered in Eulerian coordinates. On the other hand, the compartment size is set in Eulerian coordinates, and as a result, the number of Lagrangian PDE points per Eulerian compartment decreases in time.

On the growing domain, the lengths of these compartments are calculated using other simulation parameters. The length of the compartments $C_i(t)$ for $i \in \{1, ..., K(t)\}$ is given by the length of the mesoscopic part of the domain, divided by the number of compartments $K(t)$. Therefore

$$h_c(t) = \frac{L(t) - I(t)}{K(t)}. \tag{6}$$

The pseudo-compartment size is calculated from the current compartment size as follows. As well as having compartment properties we also need to solve the PDE in the region occupied by the pseudo-compartment. As such, we will set its length to be an integer number of PDE mesh points. The PDE mesh, when considered from the Eulerian perspective, has width $h_p(t) = h_p(0)\exp\{\rho t\}$, and we calculate the number of PDE mesh points that would be in the pseudo-compartment, if it was of length $h_c(t)$, to be

$$p_c(t) = \frac{h_c(t)}{h_p(t)}.$$

The value of $p_c(t)$ is generally not an integer, so in order to obtain an integer we round (up or down or to the nearest integer) the value of $p_c(t)$ and multiply this by $h_p(t)$ in order to find $h_{PC}(t)$.

If a particle is chosen to jump out of $C_1(t)$ and into $C_{-1}(t)$, we firstly reduce $n_1(t)$ by one, and then add a particle's worth of mass to the pseudo-compartment according to:

$$\boldsymbol{U}_{PC} = \boldsymbol{U}_{PC} + \frac{1}{h_{PC}(t)}\mathbf{1}, \tag{7}$$

where the vector $\boldsymbol{U}_{PC}$ is the numerical approximation to the solution $u(x,t)$ at the PDE nodes contained within the pseudo-compartment, and $\mathbf{1}$ is a vector of ones of the appropriate size. If a particle jumps out of the pseudo-compartment and into $C_1(t)$, we add one particle to $n_1(t)$ and remove a particle's worth of mass uniformly across the pseudo-compartment

$$\boldsymbol{U}_{PC} = \boldsymbol{U}_{PC} - \frac{1}{h_{PC}(t)}\mathbf{1}. \tag{8}$$

The algorithm for the implementation of the growing pseudo-compartment method (gPCM) can be found in Algorithm 1, and a schematic for the static case is given in in Figure 1. The algorithm is given for diffusion only, however reactions may be incorporated through any appropriate method (see for example (Erban and Chapman, 2009)).

Algorithm 1: The growing pseudo-compartment method (Diffusion only)

Initialise: Initial time — $t = 0$; Final time — t_f; Initial compartment size — h_c; Initial pseudo-compartment size — h_{pc}; PDE solution — $\boldsymbol{U}$; Number of pseudo-particles — n_{-1}; Number of compartments — K; Compartment particle numbers — $\boldsymbol{n}$; Propensity functions — a_i for $i \in \{1, ..., 2K+1\}$; Internal clock times — T_i for $i \in \{1, ..., 2K+1\}$; Next firing times — P_i for $i \in \{1, ..., 2K+1\}$; Times until next event — Δt_i for $i \in \{1, ..., 2K+1\}$; PDE update step — Δt; Time until next PDE update — t_p; Time until next split event — t_s; Time until next PDE re-mesh event — t_r.

(1a) At time $t > 0$:

1. Calculate $\Delta = \min\{i \in \{1, ..., 2K+1\} : \Delta t_i\}$ and $\beta = \text{argmin}\{i \in \{1, ..., 2K+1\} : \Delta t_i\}$. Set $t_\Delta = t + \Delta$.
2. If $\min\{t_\Delta, t_p, t_s, t_r\} = t_\Delta$:
   (a) For every $i \in \{1, ..., 2K+1\}$, update T_i according to
   $$T_i \leftarrow T_i + \frac{1}{2\rho} a_i \left(1 - \exp\{-2\rho\Delta\}\right).$$
   (b) For event β, set
   $$P_\beta \leftarrow P_\beta + \ln\left(\frac{1}{u_1}\right), \text{ where } u_1 \sim \text{Unif}(0, 1).$$
   (c) Enact the event β:
   - If the event β corresponds to a jump from the pseudo-compartment to the first compartment, set $n_1 \leftarrow n_1 + 1$ and set $\boldsymbol{U}_{pc} \leftarrow \boldsymbol{U}_{pc} - 1/h_{pc}\mathbf{1}$, where $\boldsymbol{U}_{pc}$ are the pseudo-compartment nodes of the PDE solution, and $\mathbf{1}$ is a vector of ones of the appropriate size.
   - If the event β corresponds to a jump from the first compartment to the pseudo-compartment, set $n_1 \leftarrow n_1 - 1$ and set $\boldsymbol{U}_{pc} \leftarrow \boldsymbol{U}_{pc} + 1/h_{pc}\mathbf{1}$.
   - Otherwise, set $\boldsymbol{n} \leftarrow \boldsymbol{n} + \boldsymbol{\nu}_\beta$, where $\boldsymbol{\nu}_\beta$ is the stoichiometric vector for the event β.

   (d) Set $t = t_\Delta$.
3. Else if $\min\{t_\Delta, t_p, t_s, t_r\} = t_s$:
   (a) Enact a growth event according to Smith et al. (2019) (see Algorithm S.4). Set $K' = K + 1$.
   (b) For every $i \in \{1, ..., 2K+1\}$, update T_i according to
   $$T_i \leftarrow T_i + \frac{1}{2\rho} a_i \left(1 - \exp\{-2\rho(t_s - t)\}\right).$$
   (c) For $i \in \{2K+2, 2K+3\}$, set $T_i = 0$ and $P_i = \ln(1/u_{2,i})$, where $u_{2,i} \sim \text{Unif}(0, 1)$.

(d) Set $t = t_s$. Update t_s.

(e) Set $K \leftarrow K'$.

4. Else if $\min\{t_\Delta, t_p, t_s, t_r\} = t_p$:

   (a) For every $i \in \{1, ..., 2K+1\}$, update T_i according to

$$T_i \leftarrow T_i + \frac{1}{2\rho} a_i \left(1 - \exp\{-2\rho(t_p - t)\}\right).$$

   (b) Enact a PDE update step using Algorithm S.1.

   (c) Set $t \leftarrow t_p$. Set $t_p \leftarrow t_p + \Delta t$.

5. Else:

   (a) For every $i \in \{1, ..., 2K+1\}$, update T_i according to

$$T_i \leftarrow T_i + \frac{1}{2\rho} a_i \left(1 - \exp\{-2\rho(t_r - t)\}\right).$$

   (b) Re-mesh the PDE solution according to Algorithm S.2

   (c) Set $h_p \leftarrow h_p/2$.

   (d) Set $t \leftarrow t_r$. Update t_r according to Algorithm S.2.

6. Update h_c and h_{pc}.

7. Update all propensity functions a_i, for $i \in \{1, ..., 2K+1\}$.

8. Update Δt_i according to

$$\Delta t_i = -\frac{1}{2\rho} \ln\left(1 - \frac{2\rho(P_i - T_i)}{a_i}\right),$$

   for $i \in \{1, ..., 2K+1\}$.

(1b) If $t < t_f$, return to **(1a)**. Otherwise, end.

4 The ghost-cell method

Within this section, we describe the growing ghost cell method (gGCM), the static counterpart of which was proposed by Flegg et al. (2015). This hybrid method couples the mesoscopic and microscopic descriptions of reaction-diffusion systems. We define the two spatial domains to be $\Omega_{\text{B}}(t) = (0, I(t))$ for the microscopic, Brownian-based dynamics, and $\Omega_{\text{C}}(t) = (I(t), L(t))$ for the compartment-based subdomain. The definitions of $I(t)$ and $L(t)$ are the same as for gPCM.

The coupling is implemented in a similar way to the PCM. A ghost cell (which is analogous to the pseudo-compartment for the PCM) is created within the microscopic subdomain, adjacent to the interface. Transport of mass over the interface is implemented using the mesoscopic approach. As such, the microscopic subdomain has a reflective boundary at the interface to ensure that no particles are able to move across through that medium.

Unlike in the PCM, since we don't have two different discretisation lengths being used in the numerical realisation of the algorithm (PDE and compartment meshes), we are free to choose the ghost-cell to be the same size as the other compartments in the simulation (as they change in size), and we set it to

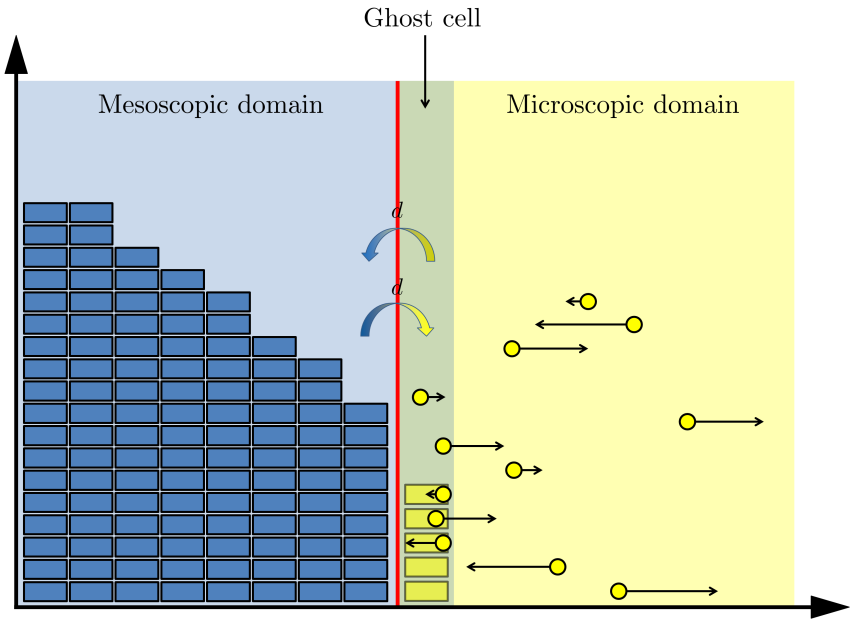


Figure 2. A schematic for the static GCM (Flegg et al., 2015). The blue rectangles and red line are the same as in Figure 1. The yellow dots, denote the positions of the individual particles, with arrows denoting the next jump size and direction. Note that we have given each particle a different height to aid clarity, but all particles lie on the axis in reality. The yellow rectangles are the number of ghost cell particles, while the arrows over the interface denote the direction of travel for the ghost cell particles. We further note that this diagram has the two subdomains in the opposite order compared to the description in the text.

$h_c(t)$ as defined in equation (6). The ghost cell therefore occupies the region $C_{\text{GC}}(t) = (I(t) - h_c(t), I(t))$. The propensity function for particles to jump from the ghost cell into the first compartment in the compartment-based regime is $D/(h_{PC}(t)^2)$, multiplied by the number of particles $n_{\text{GC}}(t)$ in the ghost cell. $n_{\text{GC}}(t)$ is calculated simply by counting the number of particles within the ghost cell, so

$$n_{\text{GC}}(t) = \sum_{i=1}^{n_{\text{B}}(t)} \mathbb{1}_{[y_i(t) \in C_{\text{GC}}(t)]},$$

where $\mathbb{1}_{[A]}$ is the indicator function that is one if A is true and 0 otherwise, $y_i(t)$ is the location of Brownian particle i at time t, and $n_{\text{B}}(t)$ is the total number of particles in the Brownian-based subdomain at time t.

When a particle jumps from the ghost cell to the first compartment, we increase $n_1(t)$ by one and remove one of the ghost cell particles uniformly at random whilst simultaneously reducing $n_{\text{GC}}(t)$ by 1. When a jump occurs from the first compartment into the ghost cell, we reduce $n_1(t)$ by one and add a new particle to the ghost cell by sampling a uniform position within the ghost cell (that is, add a new particle at a position $y^* \sim \text{Unif}(I(t) - h_c(t), I(t))$) and subsequently increase $n_{\text{GC}}(t)$ by 1. Domain growth is implemented using the stretching method (Smith et al., 2019) for the mesoscale, and via the SDE for

the microscale. The algorithm for the growing ghost-cell method can be found in Algorithm 2, and a schematic for the static case is given in Figure 2.

Algorithm 2: The growing ghost cell method (Diffusion only)

Initialise: Initial time — $t = 0$; Final time — t_f; Compartment size — h_c; Positions of particles — $\boldsymbol{y}$; Number of compartments — K; Compartment particle numbers — $\boldsymbol{n}$; Propensity functions — a_i for $i \in \{1, ..., 2K+1\}$; Internal clock times — T_i for $i \in \{1, ..., 2K+1\}$; Next firing times — P_i for $i \in \{1, ..., 2K+1\}$; Times until next event — Δt_i for $i \in \{1, ..., 2K+1\}$; Brownian update step — Δt; Time until next Brownian update — t_b; Time until next splitting event — t_s.

(2a) At time $t > 0$:

1. Calculate $\Delta = \min\{i \in \{1, ..., 2K+1\} : \Delta t_i\}$ and $\beta = \text{argmin}\{i \in \{1, ..., 2K+1\} : \Delta t_i\}$. Set $t_\Delta = t + \Delta$.
2. If $\min\{t_\Delta, t_b, t_s\} = t_\Delta$:
   (a) For every $i \in \{1, ..., 2K+1\}$, update T_i according to
   $$T_i \leftarrow T_i + \frac{1}{2\rho} a_i \left(1 - \exp\{-2\rho\Delta\}\right).$$
   (b) For event β, set
   $$P_\beta \leftarrow P_\beta + \ln\left(\frac{1}{u_1}\right), \text{ where } u_1 \sim \text{Unif}(0, 1).$$
   (c) Enact the event β:
   - If the event β corresponds to a jump from the ghost cell to the first compartment, set $n_1 \leftarrow n_1 + 1$ and remove a particle from the ghost cell uniformly at random.
   - If the event β corresponds to a jump from the first compartment to the ghost cell, set $n_1 \leftarrow n_1 - 1$ and add a new particle to the ghost cell by drawing $u_2 \sim \text{Unif}(0, 1)$ and set the new particle's position y^* to be $y^* = I(t) - u_2 h_c(t)$.

   (d) Set $t = t_\Delta$.
3. Else if $\min\{t_\Delta, t_b, t_s\} = t_s$:
   (a) Enact a growth event according to Smith et al. (2019) (see Algorithm S.4 in the SM). Set $K' = K + 1$.
   (b) For every $i \in \{1, ..., 2K+1\}$, update T_i according to
   $$T_i \leftarrow T_i + \frac{1}{2\rho} a_i \left(1 - \exp\{-2\rho(t_s - t)\}\right).$$
   (c) For $i \in \{2K+2, 2K+3\}$, set $T_i = 0$ and $P_i = \ln(1/u_{3,i})$, where $u_{3,i} \sim \text{Unif}(0, 1)$.
   (d) Set $t = t_s$. Update t_s.
   (e) Set $K \leftarrow K'$.
4. Else:

(a) For every $i \in \{1, ..., 2K+1\}$, update T_i according to

$$T_i \leftarrow T_i + \frac{1}{2\rho} a_i \left(1 - \exp\{-2\rho(t_b - t)\}\right).$$

(b) Enact a Brownian update step using algorithm S.5.

(c) Set $t \leftarrow t_b$. Set $t_b \leftarrow t_b + \Delta t$.

5. Update $h_c(t)$.
6. Update all propensity functions a_i, for $i \in \{1, ..., 2K+1\}$.
7. Update Δt_i according to

$$\Delta t_i = -\frac{1}{2\rho} \ln\left(1 - \frac{2\rho(P_i - T_i)}{a_i}\right),$$

for $i \in \{1, ..., 2K+1\}$.

(2b) If $t < t_f$, return to **(2a)**. Otherwise, end.

5 The auxiliary region method

Within this section, we describe the third and final of our growing hybrid methods, the growing auxiliary region method (gARM) (Smith and Yates, 2018a). This method couples the macro and microscales, using a similar methodology to both the PCM and the GCM. Auxiliary regions are set up on either side of the interface, which act as compartments for the purpose of allowing particles to move between the two subdomains. A schematic for the static version of the method is in Figure 3.

The two auxiliary regions are equally sized compartments (although it is possible to have them at different sizes), where particle numbers in the PDE auxiliary region are calculated as in the pseudo-compartment of the gPCM for the macroscopic subdomain, and particles in the Brownian auxiliary region are calculated as in the ghost cell of the gGCM for the microscopic domain. As is the case in each of those other hybrid methods, compartment-based jumping via the auxiliary regions is the only mechanism by which particles may pass between the subdomains. As such, the PDE requires a zero-flux boundary condition at the interface, and the Brownian-based dynamics need an equivalent reflective boundary.

In order to calculate the size of the auxiliary regions, we use a similar idea to that employed in the gPCM. We set the size of a compartment to be h initially, and use this to find the actual auxiliary region size, $h_{\text{AR}}(t)$ by firstly calculating how many PDE mesh points lie within this initial size, via:

$$p_c(t) = \frac{h}{h_p(t)},$$

where $h_p(t)$ is as in Section 3, the mesh-width in Eulerian coordinates. We convert this to an integer number of PDE mesh points by rounding it to the nearest integer, and convert it back to a length by multiplying by $h_p(t)$. Using this, we can write the regions occupied by the auxiliary regions to be $\Omega_{\text{PA}}(t) = (I(t) - h_{\text{AR}}(t), I(t))$ for the PDE auxiliary region and $\Omega_{\text{BA}}(t) = (I(t), I(t) + h_{\text{AR}}(t))$ for the Brownian-based auxiliary region.

Particle numbers for the PDE auxiliary region, $n_{\text{PA}}(t)$, and the Brownian auxiliary region, $n_{\text{BA}}(t)$, are

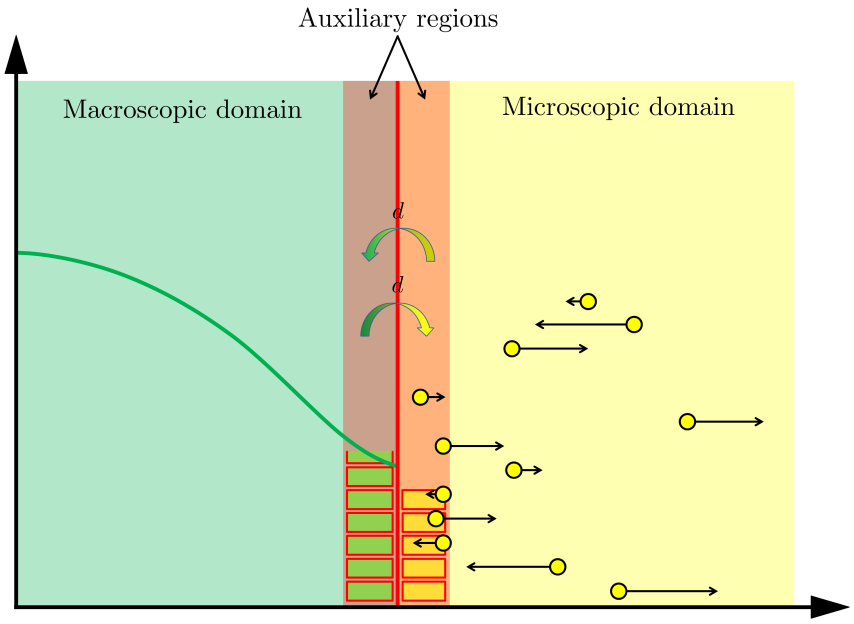


Figure 3. A schematic for the static ARM (Smith and Yates, 2018a). The green and red lines are the same as in Figure 1, while the yellow dots and arrows are the same as in Figure 2. The green and yellow rectangles denote the number of PDE and Brownian auxiliary particles respectively. The arrows over the interface denote the movement of these auxiliary particles.

calculated as in the PCM and GCM respectively, and are given by

$$n_{\mathrm{PA}}(t) = \int_{\Omega_{\mathrm{PA}}(t)} u(x,t)\, dx, \tag{9}$$

$$n_{\mathrm{BA}}(t) = \sum_{i=1}^{N_{\mathrm{B}}(t)} \mathbb{1}_{[y_i(t)\in\Omega_{\mathrm{BA}}(t)]}. \tag{10}$$

The implementation of the gARM is described in Algorithm 3 below.

Algorithm 3: The growing auxiliary region method (Diffusion only)

Initialise: Initial time — $t = 0$; Final time — t_f; PDE solution — $\boldsymbol{U}$; Positions of particles — $\boldsymbol{y}$; Propensity functions — a_i for $i = 1, 2$; Internal clock times — T_i for $i = 1, 2$; Next firing times — P_i for $i = 1, 2$; Times until next event — Δt_i for $i = 1, 2$; PDE/Brownian update step — Δt; Time until next PDE/Brownian update — t_b; Time until next re-mesh event — t_r; Auxiliary region size h_{AR}.

(3a) At time $t > 0$:

1. Calculate $\Delta = \min\{i \in \{1,2\} : \Delta t_i\}$ and $\beta = \operatorname{argmin}\{i \in \{1,2\} : \Delta t_i\}$. Set $t_\Delta = t + \Delta$.
2. If $\min\{t_\Delta, t_b, t_r\} = t_\Delta$:
   (a) For every $i \in \{1,2\}$, update T_i according to
   $$T_i \leftarrow T_i + \frac{1}{2\rho} a_i \left(1 - \exp\{-2\rho\Delta\}\right).$$
   (b) For event β, set
   $$P_\beta \leftarrow P_\beta + \ln\left(\frac{1}{u_1}\right), \text{ where } u_1 \sim \text{Unif}(0,1).$$
   (c) Enact the event β:
   - If the event β corresponds to a jump from the PDE auxiliary region to the microscopic auxiliary region, draw $u_2 \sim \text{Unif}(I(t) - h_{\text{AR}}, I(t))$ and place a new particle at that position, and set $\boldsymbol{U}_{pc} \leftarrow \boldsymbol{U}_{pc} - 1/h_{pc}\mathbf{1}$, where $\boldsymbol{U}_{pc}$ are the pseudo-compartment nodes of the PDE solution, and $\mathbf{1}$ is a vector of ones of the appropriate size.
   - If the event β corresponds to a jump from the microscopic auxiliary region to the PDE auxiliary region, choose one of the particles contained in the microscopic auxiliary region uniformly at random and remove it, and set $\boldsymbol{U}_{pc} \leftarrow \boldsymbol{U}_{pc} + 1/h_{pc}\mathbf{1}$.

   (d) Set $t = t_\Delta$.
3. Else if $\min\{t_\Delta, t_b, t_r\} = t_b$:
   (a) For $i \in \{1,2\}$, update T_i according to
   $$T_i \leftarrow T_i + \frac{1}{2\rho} a_i \left(1 - \exp\{-2\rho(t_b - t)\}\right).$$
   (b) Enact a Brownian update step using algorithm S.5.
   (c) Enact a PDE update step using algorithm S.1.
   (d) Set $t \leftarrow t_b$. Set $t_b \leftarrow t_b + \Delta t$.
4. Else:
   (a) For $i \in \{1,2\}$, update T_i according to
   $$T_i \leftarrow T_i + \frac{1}{2\rho} a_i \left(1 - \exp\{-2\rho(t_r - t)\}\right).$$
   (b) Re-mesh the PDE solution according to Algorithm S.2
   (c) Set $h_p \leftarrow h_p/2$.
   (d) Set $t \leftarrow t_r$. Update t_r according to Algorithm S.2.
5. Update h_{AR}.
6. Update all propensity functions a_i, for $i \in \{1,2\}$.
7. Update Δt_i according to
   $$\Delta t_i = -\frac{1}{2\rho} \ln\left(1 - \frac{2\rho(P_i - T_i)}{a_i}\right),$$
   for $i \in \{1,2\}$.

(3b) If $t < t_f$, return to **(3a)**. Otherwise, end.

6 Results

Within this section, we present results from three test problems, for all three of the methods described in Sections 3-5. The three test problems are designed to evaluate the performance of the algorithms in comparison to the corresponding PDE solutions. As such, the examples are relatively simple so that the PDE is in exact correspondence with the expected behaviour of the individual-based methods (i.e. no reactions of order higher than one). These choices mean that discrepancies between the mean behaviour of the hybrid methods and the PDE solutions can be be attributed directly to the hybridisation. All examples will be on a one-dimensional, exponentially growing domain, but can be straightforwardly extended to higher dimensions on Cartesian domains, and to other forms of uniform domain growth.

The next three subsections will be devoted to the three test problems which will each assess a different aspect of the performance of the three hybrid algorithms.

6.1 Test problem 1: Maintaining uniformity

The first test problem verifies that the algorithms are able to maintain a uniform particle distribution under pure diffusion. For this, we will use the growing domain diffusion equation:

$$\frac{\partial u}{\partial t}(x,t) = D\frac{\partial^2 u}{\partial x^2}(x,t) - \rho\frac{\partial (xu(x,t))}{\partial x} \qquad x \in (0, 2\exp\{\rho t\}),\ t > 0, \tag{11}$$

$$\frac{\partial u}{\partial x}(0,t) = 0 \qquad t > 0, \tag{12}$$

$$\frac{\partial u}{\partial x}(2\exp\{\rho t\}, t) = 0 \qquad t > 0 \tag{13}$$

$$u(x,0) = \frac{M}{2} \qquad x \in [0,2]. \tag{14}$$

Here, M is the number of particles in the system, and all other parameters are as in Sections 3—5. This PDE system has an analytical solution of

$$u(x,t) = \frac{M}{2}\exp\{-\rho t\} \tag{15}$$

We run this example with a diffusion coefficient of $D = 0.0025$, an exponential growth rate of $\rho = 0.001$ and $M = 500$ particles. Each hybrid simulation is averaged over 1000 independent repeats for comparison and error plotting purposes.

In Figure 4, we present the results for each of the hybrid methods (gPCM in column 1, gGCM in column 2 and gARM in column 3), with snapshots of the solution at the initial time (row 1) and final time $t_f = 500$ (row 2), and then the relative errors in the left (row 3) and right (row 4) halves of the domain. The relative error is calculated as

$$E_{\text{L}}(t) = \frac{n_{\text{L}}^H(t) - n_{\text{L}}^P(t)}{n_{\text{L}}^P(t)}, \tag{16}$$

for the left side of the domain. Here, $E_{\text{L}}(t)$ is the relative error in the left side of the domain at time t, $N_{\text{L}}^H(t)$ is the number of particles in the left subdomain of the hybrid method at time t and $n_{\text{L}}^P(t)$ is the number of particles calculated in the left-hand side of the PDE solution at time t. As can be seen from the plots, each hybrid method is able to correctly maintain uniformity with no bias in particle numbers to either subdomain.

6.2 Test problem 2: Testing flux

The second test problem is designed to assess the ability of each of the hybrid methods to cope with a non-zero flux acrosss the interface. For this, we use the same PDE from test problem 1, but change the

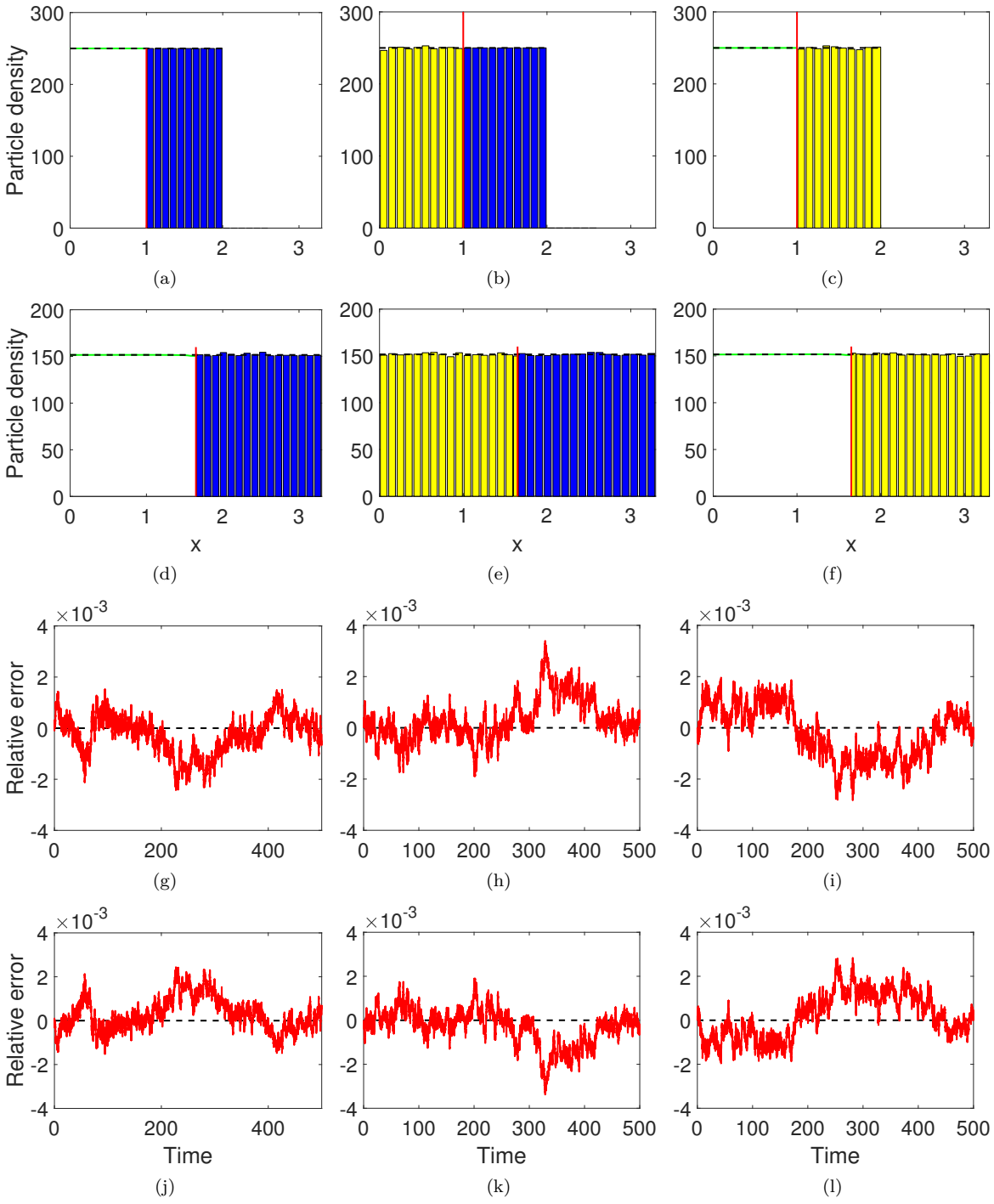


Figure 4. Results for Test Problem 1. We present results for the gPCM (Section 3) in column 1, the gGCM (Section 4) in column 2 and the gARM (Section 5) in column 3. Figures (a)-(f) are snapshots of each method at the initial (row 1) and final ($t = 500$, row 2) times of the simulation for the three methods. Green lines denote the PDE solution of the hybrid methods, blue bars are the particle densities for the mesoscale within the hybrid methods, and the microscale densities are denoted by the yellow bars, where we have binned particles onto the same mesh as the compartments. The red vertical lines on each plot denote the position of the interface at that time, and the black dashed line is the solution of the PDE across the whole domain, which we consider as our ground truth. Figures (g)-(l) display the relative errors in the left subdomain (row 3) and right subdomain (row 4) corresponding to the different modelling paradigms in the hybrid methods. The red curves denoting the relative error are given by the formula (16) for the left subdomain, with an analogous formula for the right side. The black dashed line corresponds to an error of 0. There is no bias in the error in the positive or negative direction for any of the methods.

boundary conditions to ensure a flux over the entire domain. The PDE with its boundary and initial conditions are below:

$$\frac{\partial u}{\partial t}(x,t) = D\frac{\partial^2 u}{\partial x^2}(x,t) - \rho\frac{\partial (xu(x,t))}{\partial x} \qquad x \in (0, 2\exp\{\rho t\}),\ t > 0, \tag{17}$$

$$-D\frac{\partial u}{\partial x}(0,t) = Ru(2\exp\{\rho t\}, t) \qquad t > 0, \tag{18}$$

$$-D\frac{\partial u}{\partial x}(2\exp\{\rho t\}, t) = Ru(2\exp\{\rho t\}, t) \qquad t > 0, \tag{19}$$

$$u(x,0) = \frac{M}{2} \qquad x \in [0,2]. \tag{20}$$

Equation (19) specifies a Robin boundary condition and equation (18) is the corresponding condition which ensures periodic boundaries: particles that exit the domain at the right-hand end will re-enter the domain at the left-hand end. The resultant steady state (when the domain is not growing) is a linear gradient in density.

We present our results for this test problem in Figure 5. As in the case with the first test problem, we can see that all three of the hybrid methods perform accurately with no discernible bias in particle numbers on either side of the domain. This demonstrates that the hybrid methods are able to resolve gradients over the interface.

6.3 Test problem 3: Morphogen gradient formation

For the final test problem, we investigate the formation of a morphogen gradient on a growing domain. This example is designed to determine whether the hybrid methods are able to accurately perform when zeroth- and first-order reactions are incorporated[1]. The PDE, boundary and initial conditions are below:

$$\frac{\partial u}{\partial t}(x,t) = D\frac{\partial^2 u}{\partial x^2}(x,t) - \rho\frac{\partial (xu(x,t))}{\partial x} - \mu u \qquad x \in (0, 2\exp\{\rho t\}),\ t > 0, \tag{21}$$

$$\frac{\partial u}{\partial x}(0,t) = -\lambda \qquad t > 0, \tag{22}$$

$$\frac{\partial u}{\partial x}(2\exp\{\rho t\}, t) = 0 \qquad t > 0, \tag{23}$$

$$u(x,0) = \frac{M}{2} \qquad x \in [0,2]. \tag{24}$$

The regular diffusion and dilution in the PDE is augmented with a sink term, $-\mu u$. This degradation of mass manifests as a first-order reaction of the form:

$$A \xrightarrow{\mu} \emptyset$$

in both of the mesoscopic and microscopic representations, where A represents a particle whose density is given by u. The boundary condition at the left-hand end of the domain represents an influx of particles with rate $D\lambda$. This can be thought of as a zeroth-order reaction at the left-hand boundary, of the form:

$$\emptyset \xrightarrow{\kappa} A,$$

where κ is the rate of introduction of new particles which is related to λ via $\kappa = \lambda D$.

[1]In this proof of principle paper we do not include examples of second- or higher-order interactions. We know that the mean-field PDE model does not correspond exactly to the mean of the stochastic models in these cases due to the necessity for moment closure when deriving the continuum equations from the individual-based models. To accurately determine whether our hybrd coupling introduces bias we consider only examples in which the expected behaviour of the individual-based method matches the behaviour of the equivalent continuum model.

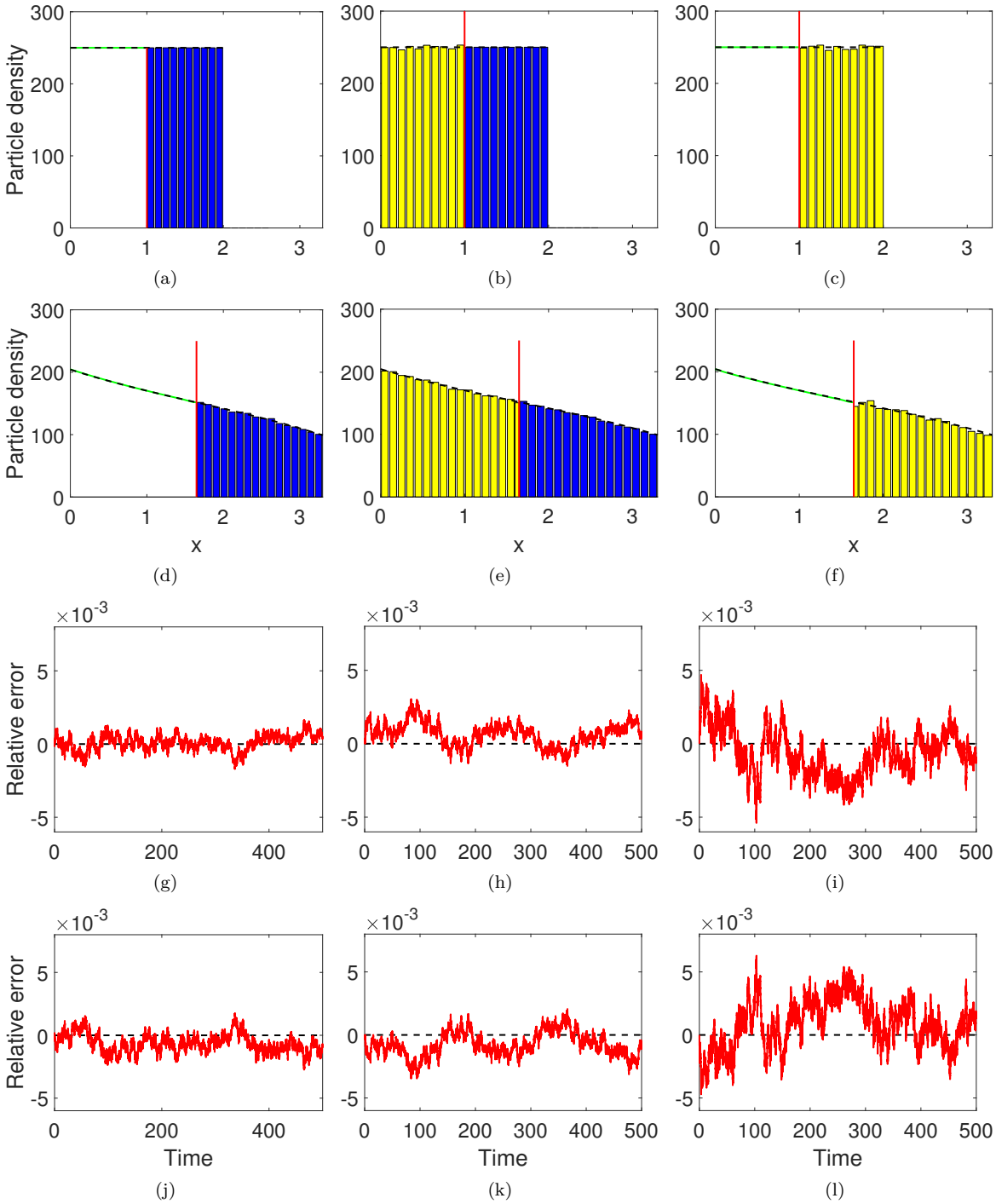


Figure 5. Results for Test Problem 2. Figure descriptions are as in Figure 4.

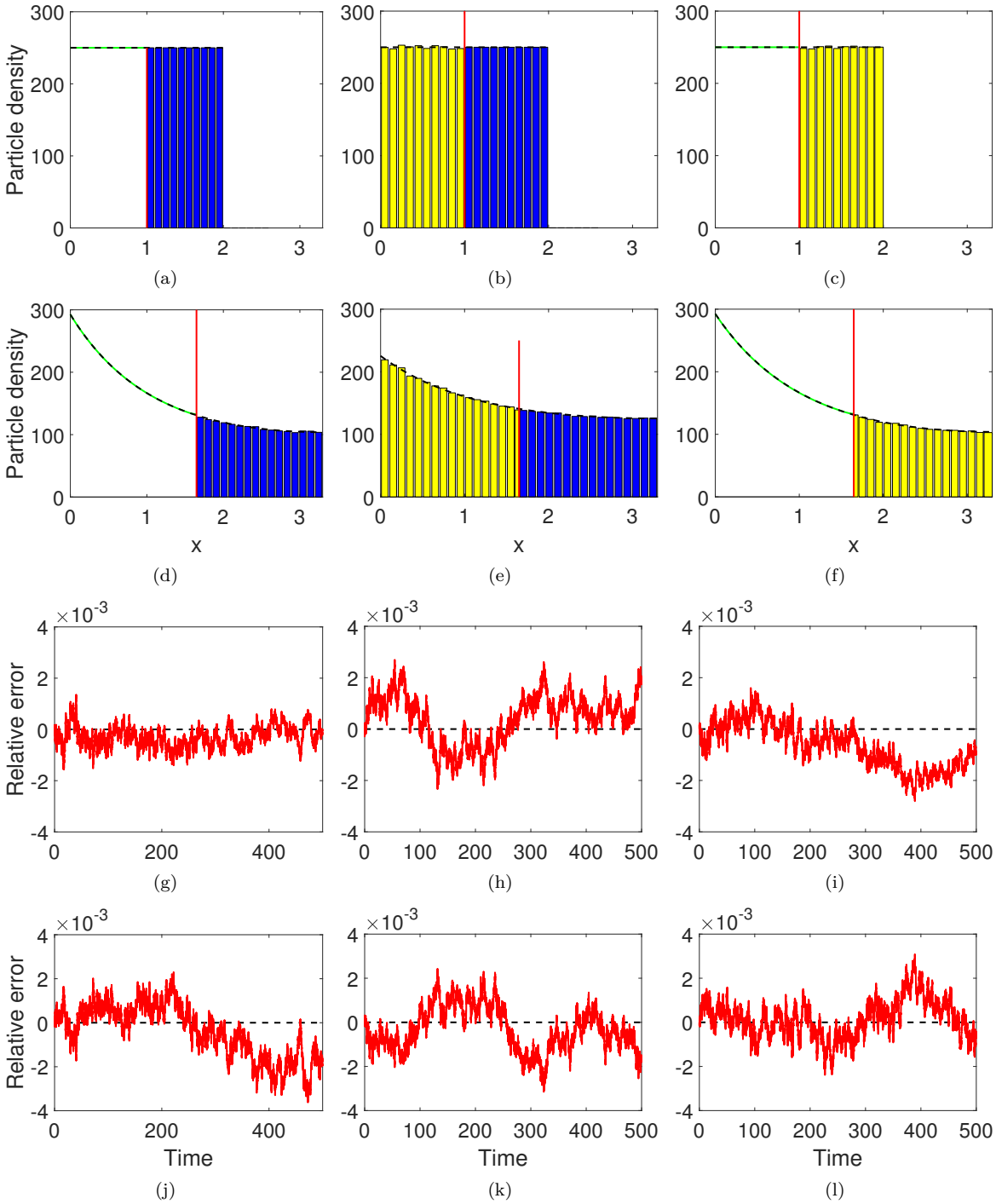


Figure 6. Results for Test Problem 3. Figure descriptions are as in Figure 4.

We present the results for test problem 3 in Figure 6. As with each of the previous test problems, we see that there is a very good qualitative agreement with all three methods and the ground truth numerical PDE solution (6(a)-6(f)) and this is further confirmed by the relative error plots, which show no discernible bias in either direction for any of the hybrid methods.

7 Discussion

In this paper, we have developed three spatially extended hybrid methods which are capable of simulating multi-scale reaction diffusion processes on uniformly growing domains. Each method has been extended from previous methods originally specified on static domains (Flegg et al., 2015; Smith and Yates, 2018a; Yates and Flegg, 2015). We have provided descriptions, schematics and detailed algorithms for the implementation of these methods. Furthermore, we have demonstrated that each of these methods are accurate and unbiased under three test problems: pure diffusion with zero-flux boundaries, diffusion with partially absorbing periodic boundaries, and the formation of a morphogen gradient with particle degradation and an influx boundary.

We have focussed on exponential growth, however it should be noted that the methods set out here will work for any type of uniform growth, provided that there is an equivalence framework between the three individual modelling paradigms. For example, if the domain were to grow linearly with rate ρ, the domain length at time t (given it is of length L_0 to begin with) would be $L(t) = L_0(1+\rho t)$. The advection term in equation (1) would become $\rho(ux)_x/(1+\rho t)$, where the subscript here denotes differentiation with respect to the x variable. In order for the mesoscale to be equivalent, domain growth events must occur with rate $\rho/(h_c K)$ (Baker et al., 2010). We must also be careful to adapt the calculation of the internal times of the mesoscopic events in each of the algorithms for this altered domain growth scenario (Anderson, 2007).

We have also presented each of our examples here in one dimension for simplicity and clarity of explanation, however, it would be straightforward to extend the methods to higher dimensions which have (hyper-)planar interfaces. Care must be taken if domain growth is to be implemented in higher dimensions, that the interface remains coherent. For example, when implementing domain growth in the compartment-based method, it makes sense to employ deterministic growth as in (Smith et al., 2019) (rather than stochastic growth (Baker et al., 2010)) so that the rate of domain growth along the interface matches in each of the coupled methodologies.

There are several extensions (many adapted from the static hybrid literature) that might be included in order to render these growing hybrid methods more versatile. The inclusion of an adaptive interface, for example, would allow the methods to more robustly and efficiently deal with examples in which the concentration profile changes significantly. Static interfaces (interfaces which remain in the same place when considered in Lagrangian coordinates) are useful when we either know something about what the solution looks like *a priori*, and hence we can place the interface in an appropriate position, or if there is a region of the spatial domain that requires a more detailed representation (such as around ion channels when considering transmembrane transport - see, for example (Dobramysl et al., 2015)). However, if neither of these are true, an adaptive interface may be required. Implementing an adaptive interface requires the use of local densities around the interface in order to determine its position; if the density in the finer scale subdomain is too high, the interface can move into that subdomain - extending the coarser subdomain, and vice-versa if the local density in the coarser subdomain is too low (Harrison and Yates, 2016; Robinson et al., 2014; Smith et al., 2019).

While domain growth is an important phenomena in many biological scenarios (Baker et al., 2010), domain shrinkage also has equally important applications. As an example, directed apoptosis is an important component of wound healing, and requires models that incorporate domain shrinkage (Yates, 2014). The hybrid methods presented in this paper would extend equally well to a domain that uniformly shrinks in size as they do to those that grow.

The hybrid methods presented here provide accurate ways of simulating reaction-diffusion systems on uniformly growing domains, and will be of interest to those who model such systems that either have a scale difference in particle numbers across their domain, or have regions of space which need to be modelled in more detail than others. The methods developed here will allow members of the modelling community to probe the important effects of stochasticity in their multiscale systems without the suffering potential simulation penalties that modelling the entire system using a fine-scale methodology might bring.

References

F.J. Alexander, A.L. Garcia, and D.M. Tartakovsky. Algorithm refinement for stochastic partial differential equations: I. linear diffusion. *J. Comput. Phys.*, 182(1):47–66, 2002.

V. Andasari, R.T. Roper, M.H. Swat, and M.A.J. Chaplain. Integrating intracellular dynamics using CompuCell3D and Bionetsolver: applications to multiscale modelling of cancer cell growth and invasion. *PLoS One*, 7(3):e33726, 2012.

D.F. Anderson. A modified next reaction method for simulating chemical systems with time dependent propensities and delays. *J. Chem. Phys.*, 127(21):214107, 2007.

A. Auger, P. Chatelain, and P. Koumoutsakos. R-leaping: Accelerating the stochastic simulation algorithm by reaction leaps. *J. Chem. Phys.*, 125:084103, 2006.

R.E. Baker, C.A. Yates, and R. Erban. From microscopic to macroscopic descriptions of cell migration on growing domains. *Bull. Math. Biol.*, 72(3):719–762, 2010.

A.J. Black and A.J. McKane. Stochastic formulation of ecological models and their applications. *Trends Ecol. Evol.*, 27(6):337–345, 2012.

S.C. Brenner and C. Carstensen. *Finite element methods*, 2004.

E.J. Crampin, E.A. Gaffney, and P.K. Maini. Reaction and diffusion on growing domains: scenarios for robust pattern formation. *Bull. Math. Biol.*, 61(6):1093–1120, 1999.

U. Dobramysl, S. Rüdiger, and R. Erban. Particle-based multiscale modeling of intracellular calcium dynamics. *Multiscale Modeling & Simulation*, 14(3):997–1016, 2015.

D. Dürr, S. Goldstein, and J.L. Lebowitz. A mechanical model of brownian motion. *Commun. Math. Phys.*, 78(4):507–530, 1981.

R. Erban. From molecular dynamics to Brownian dynamics. *Proc. R. Soc. A*, 470(2167):20140036, 2014.

R. Erban. Coupling all-atom molecular dynamics simulations of ions in water with brownian dynamics. In *Proc. R. Soc. A*, volume 472, page 20150556. The Royal Society, 2016.

R. Erban and S.J. Chapman. Stochastic modelling of reaction–diffusion processes: algorithms for bimolecular reactions. *Phys. Biol.*, 6(4):1–18, 2009.

R. Eymard, T. Gallouët, and R. Herbin. Finite volume methods. *Handbook of numerical analysis*, 7: 713–1018, 2000.

M.B. Flegg, S.J. Chapman, and R. Erban. The two-regime method for optimizing stochastic reaction–diffusion simulations. *J. Roy. Soc. Interface.*, 9(70):859–868, 2012.

M.B. Flegg, S. Hellander, and R. Erban. Convergence of methods for coupling of microscopic and mesoscopic reaction-diffusion simulations. *J. Comput. Phys.*, 289(C):1–17, 2015.

B. Franz, M.B. Flegg, S.J. Chapman, and R. Erban. Multiscale reaction-diffusion algorithms: PDE-assisted Brownian dynamics. *SIAM J. Appl. Math.*, 73(3):1224–1247, 2013.

M.A. Gibson and J. Bruck. Efficient exact stochastic simulation of chemical systems with many species and many channels. *J. Phys. Chem. A.*, 104(9):1876–1889, 2000.

D.T. Gillespie. Exact stochastic simulation of coupled chemical reactions. *J. Phys. Chem.*, 81(25): 2340–2361, 1977.

D.T. Gillespie. Approximate accelerated stochastic simulation of chemically reacting systems. *J. Chem. Phys.*, 115(4):1716–1733, 2001.

D.T. Gillespie, A. Hellander, and L.R. Petzold. Perspective: Stochastic algorithms for chemical kinetics. *J. Chem. Phys.*, 138(17):170901, 2013.

J.U. Harrison and C.A. Yates. A hybrid algorithm for coupling PDE and compartment-based dynamics. *J. Roy. Soc. Interface*, 13:20160335, 2016.

R. Holley. The motion of a heavy particle in an infinite one dimensional gas of hard spheres. *Probab. Theory. Rel.*, 17(3):181–219, 1971.

S. Khan, Y. Zou, A. Amjad, A. Gardezi, C.L. Smith, C. Winters, and T.S. Reese. Sequestration of camkii in dendritic spines in silico. *J. Comput. Neurosci.*, 31(3):581–594, 2011.

P.M. Kulesa, C.M. Bailey, J.C. Kasemeier-Kulesa, and R. McLennan. Cranial neural crest migration: new rules for an old road. *Dev. Biol.*, 344(2):543–554, 2010.

N.I. Markevich, J.B. Hoek, and B.N. Kholodenko. Signaling switches and bistability arising from multisite phosphorylation in protein kinase cascades. *J. Cell Biol.*, 164(3):353–359, 2004.

R. McLennan, L. Dyson, K.W. Prather, J.A. Morrison, R.E. Baker, P.K. Maini, and Kulesa P.M. Multiscale mechanisms of cell migration during development: theory and experiment. *Development*, 139 (16):2935–2944, 2012.

E. Moro. Hybrid method for simulating front propagation in reaction-diffusion systems. *Phys. Rev. E*, 69(6):060101, 2004.

R.L. Mort, R.J.H. Ross, K.J. Hainey, O.J. Harrison, M.A. Keighren, G. Landini, R.E. Baker, K.J. Painter, I.J. Jackson, and C.A. Yates. Reconciling diverse mammalian pigmentation patterns with a fundamental mathematical model. *Nat. Commun.*, 7:10288, 2016.

K.W. Morton and D.F. Mayers. *Numerical Solution of Partial Differential Equations*. Cambridge University Press, 2005.

H. Risken. Fokker-planck equation. In *The Fokker-Planck Equation*, pages 63–95. Springer, 1996.

M. Robinson, M.B. Flegg, and R. Erban. Adaptive two-regime method: application to front propagation. *J. Chem. Phys.*, 140(12):124109, 2014.

J.A. Sherratt. An analysis of vegetation stripe formation in semi-arid landscapes. *J. Math. Biol.*, 51(2): 183–197, 2005.

M.J. Simpson. Exact solutions of linear reaction-diffusion processes on a uniformly growing domain: Criteria for successful colonization. *PLoS One*, 10(2):e0117949, 2015.

C.A. Smith and C.A. Yates. The auxiliary region method: A hybrid method for coupling a PDE to Brownian-based dynamics for reaction-diffusion systems. *R. Soc. Open Sci.*, 5(8), 2018a.

C.A. Smith and C.A. Yates. Spatially extended hybrid methods: a review. *J. Roy. Soc. Interface*, 15 (139), 2018b.

C.A. Smith, C Mailler, and C.A Yates. Unbiased on-lattice domain growth. *Phys. Rev. E*, 100(6), dec 2019.

G.D. Smith. *Numerical solution of partial differential equations: finite difference methods*. Oxford University Press, 1985.

M. Smoluchowski. Versuch einer mathematischen theorie der koagulationskinetik kolloider lösungen. *Z. Phys. Chem.*, 92(129-168):9, 1917.

F. Spill, P. Guerrero, T. Alarcon, P.K. Maini, and H. Byrne. Hybrid approaches for multiple-species stochastic reaction–diffusion models. *J. Comput. Phys.*, 299:429–445, 2015.

P.R. Taylor, R.E. Baker, and C.A. Yates. Deriving appropriate boundary conditions, and accelerating position-jump simulations, of diffusion using non-local jumping. *Phys. Biol.*, 12(1):016006, 2015.

J.S. van Zon and P.R. ten Wolde. Green's-function reaction dynamics: a particle-based approach for simulating biochemical networks in time and space. *J. Chem. Phys.*, 123(23):234910, 2005.

V. Volpert and S. Petrovskii. Reaction–diffusion waves in biology. *Phys. Life Rev.*, 6(4):267–310, 2009.

L. Wolpert, C. Tickle, and A.M. Arias. *Principles of development*. Oxford University Press, 5 edition, 2015.

T.E. Woolley, R.E. Baker, E.A. Gaffney, and P.K. Maini. Stochastic reaction and diffusion on growing domains: understanding the breakdown of robust pattern formation. *Phys. Rev. E*, 84(4):046216, 2011.

C.A. Yates. Discrete and continuous models for tissue growth and shrinkage. *J. Theor. Biol.*, 350:37–48, 2014.

C.A. Yates and M.B. Flegg. The pseudo-compartment method for coupling partial differential equation and compartment-based models of diffusion. *J. Roy. Soc. Interface*, 12(106):20150141, 2015.

R. ZhuGe, K.E. Fogarty, R.A. Tuft, L.M. Lifshitz, K. Sayar, and J.V. Walsh. Dynamics of signaling between Ca2+ sparks and Ca2+-activated K+ channels studied with a novel image-based method for direct intracellular measurement of ryanodine receptor Ca2+ current. *J. Gen. Physiol.*, 116(6):845–864, 2000.

Supplementary Material
Incorporating domain growth into hybrid methods for reaction-diffusion systems

Cameron A. Smith[1,*], Christian A. Yates[1]
[1]**Centre for Mathematical Biology, Department of Mathematical Sciences, University of Bath, Claverton Down, Bath, BA2 7AY, United Kingdom**

Key words: Reaction–diffusion, domain growth, hybrid methods.

In this document, we introduce any of the mathematics required to demonstrate equivalence between the three methods described in Section 2, and give the numerical algorithms that we employ. It should be noted that other numerical schemes may be used.

S.1 Macroscale

S.1.1 The Lagrangian PDE

We wish to be able to simulate the solution of the PDE (1), with appropriate boundary and initial conditions. However, conventional techniques are unable to do so when the PDE is written in this form due to the existence of the growing domain [Simpson, 2015]. Instead, we transform the PDE onto a fixed coordinate system, known as the Lagrangian coordinates. We apply the following change of variables:

$$x = X\exp\{\rho\tau\}, \tag{S.1}$$

$$t = \tau, \tag{S.2}$$

where $X \in (0, L_0)$ and $\tau > 0$. Applying change of variables (S.1)-(S.2) to the Eulerian PDE (1) yields the Lagrangian PDE

$$\frac{\partial u}{\partial \tau}(X,\tau) = \frac{D}{\exp\{2\rho\tau\}}\frac{\partial^2 u}{\partial X^2}(X,\tau) - \rho u(X,\tau). \tag{S.3}$$

The domain growth manifests itself in equation S.3 in two ways: firstly by creating a time-dependent diffusion coefficient, which comes from the conversion of the second order derivative; and secondly by converting the dilution from an advective term to a particle sink via a first-order degredation reaction. This second term is one of the two terms obtained when the dilution term in (1) is expanded using the product rule. The second of these product rule terms cancels with a term that originates from the conversion of the time derivative.

S.1.2 Numerical scheme

In order to simulate the PDE (S.3), we will use the θ-method (see, for example [Smith, 1985]), however we note that any different simulation methodologies may be appropriate. The θ-method involves writing the second derivative as a combination of implicit and explicit terms, and using a time-stepping algorithm in order to find the solution at a later time. a description can be found in Algorithm S.1 (below) for the PDE (S.3) together with zero-flux boundary conditions. Because the PDE contains time-varying coefficients, the matrices used in order to update the solution need to be calculated at every time-step.

Algorithm S.1: The θ-method (diffusion only)

Input: Current PDE solution vector — $\boldsymbol{U}$; Time of previous update — t_n; Time-step — Δt; θ-value — θ; Diffusion coefficient D; Lagrangian mesh size — h_p; Number of mesh points — $S+1$; Growth rate — ρ; Initial domain length L_0.

(1a) Calculate the time-dependent matrices A_n and B_n:

$$(A_n)_{ij} = \begin{cases} 1 + \frac{D\Delta t\theta}{h_p^2}\exp\{-2\rho(t_n+\Delta t)\} & i=j=1, J+1 \\ 1 + 2\frac{D\Delta t\theta}{h_p^2}\exp\{-2\rho(t_n+\Delta t)\} & i \in \{2,...,S\}, j=i \\ -\frac{D\Delta t\theta}{h_p^2}\exp\{-2\rho(t_n+\Delta t)\} & i \in \{2,...,S+1\}, j=i-1 \\ -\frac{D\Delta t\theta}{h_p^2}\exp\{-2\rho(t_n+\Delta t)\} & i \in \{1,...,S\}, j=i+1 \\ 0 & \text{otherwise} \end{cases}$$

$$(B_n)_{ij} = \begin{cases} 1 - (\rho+\mu)\Delta t - \frac{D\Delta t(1-\theta)}{h_p^2}\exp\{-2\rho t_n\} & i=j=1, S+1 \\ 1 - (\rho+\mu)\Delta t - 2\frac{D\Delta t(1-\theta)}{h_p^2}\exp\{-2\rho t_n\} & i \in \{2,...,S\}, j=i \\ -\frac{D\Delta t(1-\theta)}{h_p^2}\exp\{-2\rho t_n\} & i \in \{2,...,S+1\}, j=i-1 \\ -\frac{D\Delta t(1-\theta)}{h_p^2}\exp\{-2\rho t_n\} & i \in \{1,...,S\}, j=i+1 \\ 0 & \text{otherwise} \end{cases}$$

(1b) Update the PDE solution $\boldsymbol{U}$ according to the following:

$$\boldsymbol{U} \leftarrow A_n^{-1} B_n \boldsymbol{U}.$$

S.1.3 Remeshing

Due to the difference in coordinate systems described in Section 3 of the main text, the solution vector is calculated on a static Lagrangian mesh. Considered in Eulerian coordinates, however, this mesh grows. As such, it is prudent to remesh the solution vector when the PDE mesh grows by a certain amount in Eulerian coordinates. We present the remeshing algorithm which requires the pre-calculation of the times that we remesh. A natural time to remesh is when the domain has doubled in length, and this is indeed how we calculate these remeshing times. However, alternative scale factors may be used.

We begin by calculating the number of times the domain doubles in length before the final time of the simulation, t_f, via

$$s_{max} = \lfloor \exp\{\rho t_f\}/2 \rfloor, \tag{S.4}$$

where $\lfloor \cdot \rfloor$ denotes the floor function. Then for $i \in \{1,...,s_{max}\}$, the ith time of remeshing is given by solving

$$2iL_0 = L_0 \exp\{\rho t_i\}. \tag{S.5}$$

The full algorithm for remeshing can be found in Algorithm S.2

Algorithm S.2: Remeshing the PDE

Input: Current PDE solution vector — $\boldsymbol{U}$; Growth rate — ρ; Current number of PDE mesh points — $S+1$; Current remesh time — t_i; Next remesh time — t_{i+1}.

(2a) Create a temporary solution vector $\boldsymbol{V}$ of size $2S+1$.

(2b) Set $V_{2j-1} = U_j$ for every $j \in \{1, ..., S+1\}$.

(2c) For all V_j with $j \in \{2, 4, 6, ..., 2S\}$, interpolate the solution. Any reasonable interpolation may be used, however for the purposes of this paper we use linear interpolation:

$$V_j = \frac{V_{j-1} + V_{j+1}}{2}, \quad j \in \{2, 4, 6, ..., 2S\}$$

(2d) Set $\boldsymbol{U} \leftarrow \boldsymbol{V}$.

(2e) Update the number of mesh points to be $2S+1$.

(2f) Update the remesh time to be t_{i+1}.

S.2 Mesoscale

S.2.1 Modified next reaction method

We will evolve the mesoscopic system according to the modified next reaction method [Anderson, 2007]. We choose to use the modified next reaction method as opposed to the standard Gillespie algorithm [Gillespie, 1977] because the propensity functions in the hybrid methods in the main text depend explicitly on time, meaning that between two events taking place, the propensity function is changing. This violates the assumption of the Gillespie algorithm, that requires the propensity functions to remain constant between the times that events take place.

In order to use the modified next reaction method, we require the definition of propensity functions $a_j(\boldsymbol{n}, t)$, where $j \in \{1, ..., 2K(t)\}$ indexes the possible events that can occur, with the first $2K(t)$ being the diffusive jumps left and right from each compartment. The probability of a particular event j occurring during a small time interval $(t, t+\delta t)$ is then defined to be $a_j(\boldsymbol{n}(t), t)\delta t$.

The modified next reaction method requires the calculation and tracking of internal clock times T_i for every possible event, and next firing times P_i. These are initialised as $T_i = 0$ and $P_i = \ln(1/r_i)$, where r_i is a uniformly distributed random variable between zero and one. Propensity functions a_i are initialised, and the absolute time until the next event of type i fires can be calculated by solving:

$$P_i - T_i = \int_t^{t+\Delta t_i} a_i(\boldsymbol{n}(t), s)\ ds, \tag{S.6}$$

for Δt_i at $t = 0$.

The algorithm then locates the event which happens first, which is equivalent to finding the minimum of the Δ_i, which we call Δ. This event is enacted, all internal clock times are updated according to

$$T_i \leftarrow T_i + \int_t^{t+\Delta} a_i(\boldsymbol{n}(t), s)\ ds, \tag{S.7}$$

and for the event that fires, say event β, a new next firing time is found by setting

$$P_\beta \leftarrow P_\beta + \ln\left(\frac{1}{r_\beta}\right), \tag{S.8}$$

where r_β is a uniformly distributed random variable between zero and one. Time is updated to be $t + \Delta$ and then the algorithm repeats. The time-varying nature of the propensity functions is accounted for by using the internal clocks as opposed to absolute time. The algorithm for the modified next reaction method can be found in Algorithm S.3.

Algorithm S.3: Modified next reaction method [Anderson, 2007] (diffusion only)

Input: Current mesoscopic state — $\boldsymbol{n}$, Current time — t; Final time — t_f; Current number of compartments — K; Internal clock times — T_i for $i \in \{1, ..., 2K\}$; Next firing times — P_i for $i \in \{1, ..., 2K\}$; Propensity function — a_i for $i \in \{1, ..., 2K\}$; Times until next event — Δt_i for $i \in \{1, ..., 2K\}$ (see equation (S.6)).

(3a) Find $\Delta = \min\{i \in \{1, ..., 2K\} : \Delta t_i\}$ and $\beta = \text{argmin}\{i \in \{1, ..., 2K\} : \Delta t_i\}$.

(3b) Enact the event β.

(3c) Update the internal clocks according to equation (S.7).

(3d) For event β, update the next firing time using equation (S.8).

(3e) Recalculate the propensity functions at the new time.

(3f) Update absolute time $t = t + \Delta$.

(3g) If $t < t_f$, return to step **(S.3a)**, otherwise end.

S.2.2 Domain stretching

Next we address how to stretch the domain, which we do according to the stretching method [Smith et al., 2019]. Suppose we have a state $\boldsymbol{N}^{K-1}(t)$ which contains $K(t) - 1$ compartments, and we wish to extend the domain by a single compartment. In order to do this, we will define the pre-growth state $\boldsymbol{N}^{K-1}(t) = \boldsymbol{r} \in \mathbb{R}^{K(t)-1}$ and the post-growth state to be $\boldsymbol{N}^{K}(t)\boldsymbol{n} \in \mathbb{R}^{K(t)}$. In order to determine how the particles in the pre-growth state should be redistributed to the post-growth state, we will calculate "overlap regions". Consider the compartments being of length h_c. When there are $K(t)$ compartments, we have a domain of length $K(t)h_c$. We now stretch our pre-growth state to be of the same length, meaning that each of the $K(t) - 1$ compartments are of length $h_c K(t)/(K(t) - 1)$, yielding a domain of length $K(t)h_c$. From these, we can calculate the length of the i^{th} pre-growth compartment that overlaps the i^{th} post-growth compartment. These are the overlap regions. Using this set up, it can be calculated that the i^{th} overlap region, with $K(t) - 1$ pre-growth compartments, is of size

$$\delta_i^{K(t)-1} = \frac{K(t) - i}{K(t)}, \tag{S.9}$$

which holds for $i \in \{1, ..., K(t) - 1\}$.

These overlap regions are treated as the probability of placing a particle from pre-growth compartment i, into post-growth compartment i, independent of all others. Therefore, for each pre-growth compartment, draw a Binomially distributed random variable b_i with r_i trials and probability of success $\delta_i^{K(t)-1}$.

Then for each post-growth compartment $j \in \{1, ..., K(t)\}$ we set:

$$n_j = \begin{cases} b_1, & \text{if } j = 1, \\ b_j + (r_{j-1} - b_{j-1}), & \text{if } j \in \{2, ..., K(t) - 1\}, \\ r_{K(t)-1} - b_{K(t)-1}, & \text{if } j = K(t). \end{cases} \tag{S.10}$$

We then set the new current state $\boldsymbol{N}(t)$ to be $\boldsymbol{n}$. This algorithm can be found in Algorithm S.4.

Algorithm S.4: The stretching method [Smith et al., 2019]

Input: Current mesoscopic state — $\boldsymbol{n}$; Current number of compartmemts — K.

(4a) Define the pregrowth state to be $\boldsymbol{r} \leftarrow \boldsymbol{n}$.

(4b) Calculate the overlap proportions $\delta_i^K = (K + 1 - i)/(K + 1)$.

(4c) Draw K binomial random variables $b_i \sim \text{Bin}(r_i, \delta_i^K)$.

(4d) Create an extra compartment at the right end of the postgrowth domain (increasing K by 1 by setting $K \leftarrow K + 1$).

(4e) For $j \in \{1, ..., K\}$, set n_j according to (S.10)

We can incorporate the stretching into the overall mesoscopic simulation algorithm in two ways. The first is to add an extra propensity function to the list which represents the stretching event. Over multiple repetitions of the algorithm, this will yield domains of different lengths due to the stochasticity in the number of times this event will fire. However, in the limit as the number of repeats tends to infinity, the length of the domain will become the average length of the domain, which will be the length of the equivalent PDE domain, i.e. $L(t) = L_0 \exp\{\rho t\}$. We will not take this approach, as we wish to assess the accuracy of the hybrid method, which is aided by the removal of as many sources of statistical error from other parts of the algorithm as possible. Alternatively, we use the deterministic length $L(t) = L_0 \exp\{\rho t\}$, to calculate the times at which, on average, the number of compartments should increase by 1, and will always enact a stretching event at these times.

S.2.3 Equivalence

To demonstrate equivalence of this mesoscopic scheme to the Eulerian PDE, we need to formulate the equations for the evolution of the mean number of particles in the mesoscopic description, which we do by firstly writing down the master equation for the system. Define $p(\boldsymbol{n}, k, t)$ to be the probability that the state variable $\boldsymbol{N}^K(t)$ is $\boldsymbol{n}$ and the number of compartments $K(t)$ is k at time t. The master equation describes the evolution of this probability. Then the rate of change of this probability is

$$\begin{aligned} \frac{\partial p}{\partial t}(\boldsymbol{n}, k, t) = d \sum_{i=1}^{k-1} \left[(n_i + 1) p(J_i^+ \boldsymbol{n}, k, t) - n_i p(\boldsymbol{n}, k, t) \right] + d \sum_{i=2}^{k} \left[(n_i + 1) p(J_i^- \boldsymbol{n}, k, t) - n_i p(\boldsymbol{n}, k, t) \right] \\ + \rho(k-1) \sum_{\boldsymbol{r} \in \mathcal{M}_{k-1}^M} \left[p(\boldsymbol{r}, k-1, t) \pi(\boldsymbol{n}|\boldsymbol{r}) \right] - \rho k p(\boldsymbol{n}, k, t). \end{aligned} \tag{S.11}$$

Here, we have that d is the rate for any particle to jump from its compartment to one of its neighbours, $J_i^{\pm}$ are operators which move a single particle from compartment i to compartment $i \pm 1$, then set $\mathcal{M}_k^M = \left\{ \boldsymbol{m} \in \mathbb{N}^k : \sum_{i=1}^k m_i = M \right\}$ is the set of all state variables with k compartments and M total particles, and $\pi(\boldsymbol{n}|\boldsymbol{r})$ is the transition probability from state $\boldsymbol{N}^{k-1} = \boldsymbol{r}$ to $\boldsymbol{N}^k = \boldsymbol{n}$. The first two sums on the right-hand side capture the impact of the diffusive jumps right and left respectively, and the final two terms correspond to the effects of domain growth.

In order to obtain the mean equations, we define

$$\bar{M}_i^k(t) = \sum_{\boldsymbol{n} \in \mathcal{M}_k^M} n_i p(\boldsymbol{n}, k, t). \tag{S.12}$$

to be the average number of particles in compartment i when there are k compartments in total, at time t. We can then calculate the mean equations by multiplying equation (S.11) by n_i and summing over the state space $\boldsymbol{n}$ to give:

$$\frac{\mathrm{d}\bar{M}_i^k}{\mathrm{d}t} = d\bar{M}_{i-1}^k - (2d + \rho k)\bar{M}_i^k + d\bar{M}_{i+1}^k + \rho(k-1)\left[\left(1 - \frac{k-i+1}{k}\right)\bar{M}_{i-1}^{k-1} + \frac{k-i}{k}\bar{M}_i^{k-1}\right]. \tag{S.13}$$

In order to write this as a continuous system, we let $x = ih_c$, and suppose that $\bar{M}_i^k(t) \approx u(x,t)$ and $\bar{M}_{i\pm1}^k(t) \approx u(x \pm h_c, t)$. We will finally assume that mass spreads uniformly when it grows, so that $(k-1)\bar{M}_i^{k-1} = k\bar{M}_i^k$. Substituting this into the mean equations (S.13) yields

$$\begin{aligned}\frac{\partial u}{\partial t}(x,t) \approx{}& du(x-h_c,t) - 2du(x,t) + du(x+h_c,t)) \\ &+ \rho\frac{k(i-1)}{k}u(x-h_c,t) + \rho\frac{k(k-i)}{k}u(x,t) - \rho k u(x,t).\end{aligned}$$

We apply a second order Taylor expansion about x and rearrange in derivatives of u. We also omit all arguments of the function u as all will be evaluated at (x,t). This gives us the following:

$$\begin{aligned}\frac{\partial u}{\partial t} &\approx d\left[u - h_c\frac{\partial u}{\partial x} + \frac{h_c^2}{2}\frac{\partial^2 u}{\partial x^2}\right] - 2du + d\left[u + h_c\frac{\partial u}{\partial x} + \frac{h_c^2}{2}\frac{\partial^2 u}{\partial x^2}\right] \\ &\qquad + \rho(i-1)\left[u - h_c\frac{\partial u}{\partial x} + \frac{h_c^2}{2}\frac{\partial^2 u}{\partial x^2}\right] + \rho(k-i)i - \rho k u, \\ &= \left[d - 2d + d + \rho(i-1) + \rho(k-i) - \rho k\right]u + h_c\left[-d + d - \rho(i-1)\right]\frac{\partial u}{\partial x} \\ &\qquad + \frac{h_c^2}{2}\left[d + d + \rho(i-1)\right]\frac{\partial^2 u}{\partial x^2}, \\ &= -\rho u - \rho h_c(i-1)\frac{\partial u}{\partial x} + dh_c^2\frac{\partial^2 u}{\partial x^2} + \frac{\rho}{2}h_c^2(i-1)\frac{\partial^2 u}{\partial x^2}.\end{aligned}$$

We utilise the fact that $ih_c = x$ in order to simplify this further:

$$\frac{\partial u}{\partial t} = -\rho u - \rho x\frac{\partial u}{\partial x} + dh_c^2\frac{\partial^2 u}{\partial x^2} + h_c\left[\rho\frac{\partial u}{\partial x} + \frac{\rho}{2}(x - h_c)\frac{\partial^2 u}{\partial x^2}\right].$$

We now take the diffusive limit, by taking compartment size, h_c to 0 while making the jump rate, d, infinite and keeping dh_c^2 constant. We also recognise the first two terms as being an expansion of the derivative of $-\rho u x$ to obtain

$$\frac{\partial u}{\partial t} = -\rho\frac{\partial(ux)}{\partial x} + dh_c^2\frac{\partial^2 u}{\partial x^2}. \tag{S.14}$$

Comparing this with equation (1), we note that equivalence requires the Fickian diffusion coefficient D to be related to the mesoscopic jump rate d via $D = dh_c^2$.

S.3 Microscale

S.3.1 Numerical scheme

In order to simulate the SDE (4), we utilise the Euler-Maruyama method. Suppose we have a system containing N particles, whose positions at time t are given by $y_i(t)$. Then the Euler-Maruyama method allows us to update the positions in the time interval $(t, t+\Delta t)$ according to

$$y_i(t+\Delta t) = y_i(t) + \rho y_i(t)\Delta t + \sqrt{2D\Delta t}\ \xi_i, \tag{S.15}$$

where the ξ_i are normally distributed random variables with zero mean and unit variance. The algorithm for the movement of particles can be found in Algorithm S.5

Algorithm S.5: Individual-based update (diffusion only)

Input: Current positions of particles — $\boldsymbol{y}$; Current time — t; Time-step to evolve particles — Δt; Diffusion coefficient — D; Growth rate — ρ;

(5a) Update the particles' positions according to the computational SDE (S.15).

(5b) For every y_i such that $y_i < X_0 \exp\{\rho(t+\Delta t)\}$, set

$$y_i \leftarrow y_i + 2(X_0 \exp\{\rho(t+\Delta t)\} - y_i).$$

(5c) For every $y_i > X_1 \exp\{\rho(t+\Delta t)\}$, set

$$y_i \leftarrow y_i - 2(y_i - X1 \exp\{\rho(t+\Delta t)\}).$$

We now briefly discuss how reactions are to be enacted. Zeroth-order reactions, where particles appear without the influence of any other particle, will occur with a probability $\kappa_0\Delta t$ over a time interval $(t, t+\Delta t)$, where κ_0 is the rate per unit time of the reaction. First-order reactions depend on the particles already present in the system. For every particle, the reaction is enacted with a probability $\kappa_1\Delta t$, where again, the κ_1 is the rate for each particle to react. Second-order reactions involve the interaction of two particles that are "close enough" to react. There are several methods that are able to resolve these interactions, such as the λ-ρ method [Erban and Chapman, 2009]. We refer to those interested in such methods to the following references [Smoluchowski, 1917, Andrews and Bray, 2004, Erban and Chapman, 2009, Lipková et al., 2011, van Zon and ten Wolde, 2005].

S.3.2 Equivalence

Finally, to establish equivalence with the macroscale, we define the probability of finding a particle located at a position x at time t, given that it was at a position y at time $s < t$ to be $q(x, t|y, s)$. Then, through the use of an intermediate point, we can write the Chapman-Kolmogorov equation

$$q(z, t+\Delta t|y, s) = \int_{-\infty}^{\infty} q(z, t+\Delta t|x, t) q(x, t|y, s)\ dx.$$

This equation holds for any Δt, not necessarily small. We can multiply this equation by a smooth test function $\varphi(z)$, integrate over z and relabel the integral on the LHS to be an integral with respect to x

instead of z to yield

$$\int_{-\infty}^{\infty} q(x, t+\Delta t|y, s)\varphi(x)\ dx = \int_{-\infty}^{\infty} \left[\int_{-\infty}^{\infty} q(z, t+\Delta t|x, t)\varphi(z)\ dz\right] q(x, t|y, s)\ dx. \tag{S.16}$$

Taylor expanding $\varphi(z)$ on the RHS about x to second order, we obtain three integrals which can be interpreted as the zeroth- first- and second-order moments of the distribution $q(\cdot, t+\Delta t|z, t)$. These quantities are found straightforwardly using the SDE (4). Once substituted into equation (S.16), all derivatives on the φ function are transferred to the q function, and a rearrangement yields the Eulerian PDE (1), known in this context as the Fokker-Planck equation.

References

D.F. Anderson. A modified next reaction method for simulating chemical systems with time dependent propensities and delays. *J. Chem. Phys.*, 127(21):214107, 2007.

S.A. Andrews and D. Bray. Stochastic simulation of chemical reactions with spatial resolution and single molecule detail. *Phys. Biol.*, 1(3-4):137–151, 2004.

R. Erban and S.J. Chapman. Stochastic modelling of reaction–diffusion processes: algorithms for bimolecular reactions. *Phys. Biol.*, 6(4):1–18, 2009.

D.T. Gillespie. Exact stochastic simulation of coupled chemical reactions. *J. Phys. Chem.*, 81(25): 2340–2361, 1977.

J. Lipková, K.C. Zygalakis, S.J. Chapman, and R. Erban. Analysis of Brownian dynamics simulations of reversible bimolecular reactions. *SIAM J. Appl. Math.*, 71(3):714–730, 2011.

M.J. Simpson. Exact solutions of linear reaction-diffusion processes on a uniformly growing domain: Criteria for successful colonization. *PLoS One*, 10(2):e0117949, 2015.

C.A. Smith, C Mailler, and C.A Yates. Unbiased on-lattice domain growth. *Phys. Rev. E*, 100(6), dec 2019.

G.D. Smith. *Numerical solution of partial differential equations: finite difference methods*. Oxford University Press, 1985.

M. Smoluchowski. Versuch einer mathematischen theorie der koagulationskinetik kolloider lösungen. *Z. Phys. Chem.*, 92(129-168):9, 1917.

J.S. van Zon and P.R. ten Wolde. Green's-function reaction dynamics: a particle-based approach for simulating biochemical networks in time and space. *J. Chem. Phys.*, 123(23):234910, 2005.

6.2 Conclusions

In this paper, we have developed three spatially extended hybrid methods on a uniformly growing domain, extended from the pseudo-compartment method (Yates and Flegg, 2015), the ghost cell method (Flegg et al., 2015) and the auxiliary region method (Smith and Yates, 2018). We have shown that each of these methods are accurate and unbiased under a selection of representative test problems.

The methods presented in this paper are the first of their type to be created. They will provide practical tools for anybody, be they mathematicians, physicists, chemists or biologists, who require modelling techniques for reaction-diffusion systems on growing domains. In particular, if the systems to be modelled exhibit large spatial disparities in particle numbers, the methods set out in this chapter will be of importance.

It should be noted that there are a couple of papers which have extended hybrid methods onto growing domains, however, they are of a different type of hybrid method altogether. One such paper is presented by McLennan et al. (2012) in which the authors introduce a hybrid method on a uniformly growing domain in order to investigate the movement and proliferation of neural crest cells during embryonic development, but is not a spatially extended hybrid method. Instead, hybridisation in this context refers to the fact that species are represented using different modelling paradigms. Specifically, this paper simulates the diffusion and consumption of a chemoattractant using a PDE, and the movement and proliferation of the neural crest cells themselves using a particle-based approach. The authors' findings demonstrate that, in order for the neural crest cells to fully populate the growing domain, it is necessary to have at least two types of neural crest cell: those that are the first to explore space, coined as the "leaders", and those which come after, known as the "followers".

In order to couple the macroscopic chemoattractant to the particle-based neural crest cells, there are specific rules which allow interaction between the two representations. Firstly, the local chemoattractant gradient around each particle dictates the direction of movement of each neural crest cell. If a particle senses a high enough chemoattractant gradient, it will move down that gradient. In contrast, the chemoattractant is being produced, diffusing and being diluted by the growth mechanism. It is also consumed by each of the neural crest cells. This is achieved by incorporating a degradation term in the PDE, which removes chemoattractant in a small region around each neural crest cell.

While this is technically known as a hybrid method on a growing domain, it is not the spatially extended type of method that we deal with primarily in this thesis. In particular, it is not designed for coupling different modelling regimes in distinct regions of space in order to deal with disparities in density. For example, if there is need to simulate a travelling wave on a growing domain. Under such circumstances, the spatial hybrid methods developed in this paper will provide an accurate and robust method in order to simulate systems that fall into this category.

The unanswered questions that have arisen from this paper are very similar to those that occur when considering spatially extended hybrid methods on a static domain, and broadly fall into two categories: adaptive interfaces and correction of variances. We will go through each of these in more detail in the next discussion which follows this chapter.

In the context of the thesis, this final chapter of new work brings together elements from each of the previous chapters. Chapters 2 and 3 lay the groundwork for the creation of spatially extended hybrid methods, particularly those that are developed in this chapter. In Chapter 4 we create a new algorithm for the growing mesoscale which is utilised for both the gPCM and gGCM. Finally, in Chapter 5, an equivalence framework for a set of Robin boundary conditions is formulated, allowing us to accurately test our spatially extended hybrid methods on growing domains.

Chapter 7

Final conclusions and outlook

The main objective of the work presented in this thesis was to create three spatially extended hybrid methods on uniformly growing domains. In order to achieve this objective we firstly had to create a macroscopic-to-microscopic method that is both accurate and robust to parameter changes. We then required two equivalence frameworks on growing domains. The first was a new mesoscopic method for extending a spatial domain without introducing particle bias. The second required us to calculate local rates and probabilities in order to match a general boundary condition.

In Chapter 2 we introduced spatially extended hybrid methods through a comprehensive review of the literature, together with worked examples, algorithms and accompanying code. In Chapter 3, we developed a macroscopic-to-microscopic hybrid method, called the auxiliary region method, which is robust to parameter changes and is able to replicate a wide range of test problems involving zeroth-, first- and second-order reactions, an adaptive interface and problems in higher dimensions. It was developed as an alternative to a previous method (Franz et al., 2013) which we demonstrate is too sensitive to changes in parameters to make it a robust choice of hybrid method.

In Chapters 4 and 5, we addressed the two equivalence frameworks related to reaction-diffusion systems. Chapter 4 was concerned with creating a new method of domain growth for the mesoscale, which we call the stretching method. We showed that the method was able to correctly match its corresponding PDE without causing an artificial build up of particles close to the boundary — a phenomenon that was present in Baker et al. (2010). This paper is presented as a method which is more applicable to a wider range of problems than that of Baker et al. (2010), however, we also investigate the

parameters in which this original method is sufficiently accurate to be acceptable.

We then studied a second equivalence framework in Chapter 5 for a general Robin boundary condition on uniformly growing domains, relating the two stochastic representations to their PDE counterpart, and further demonstrated that all four methods match through three example simulations. This work comprised two proofs based on papers by Erban and Chapman (2007) and Singer et al. (2008). The equivalence of boundary conditions was employed when testing the accuracy of the growing hybrid methods developed in Chapter 6.

Each of the previous chapters are brought together in order to create three spatially extended hybrid methods on growing domains, in Chapter 6. We extend the ARM, together with the pseudo-compartment method (Yates and Flegg, 2015) and the ghost cell method (Flegg et al., 2015) onto uniformly growing domains — the first time that spatially adapted hybrid methods have been extended in such a way. We demonstrate that each of the three methods are able to correctly simulate multiple example problems, including examples with reactions.

While the work carried out in this thesis provides valuable methods for the mathematical and biological communities, there are still several avenues for future work which will improve the applicability to real world biological problems. In the next section, we detail some of these, and how they would improve any existing methods.

7.1 Future research

Variance correction

Many hybrid methods focus on replicating the correct mean-field behaviour for the system in question, showing that on average, the hybrid method is able to accurately match the correct behaviour. However, an important consideration may be to replicate the variation in the solution too. The mesoscale and the microscale naturally have variation associated with them, whereas the macroscopic PDE does not. As a result, any spatial coupling that involves the macroscale will have the incorrect variance in a large part of the domain. Moreover, due to the lack of variation in the PDE, the variance close to the interface in the stochastic subdomain will also be reduced.

A couple of suggestions have been made in the literature to fix this problem. The first

is to utilise an overlap region instead of an interface (Flekkøy et al., 2001; Franz et al., 2013; Harrison and Yates, 2016; Yates et al., 2020). These overlap regions are able to move the damping effect caused by the PDE away from the interface that connects the pure stochastic regime and the overlap region. This means that the variance is largely corrected in the purely stochastic regime in comparison to an interfacial method.

The second method is to replace the PDE with an appropriate reaction-diffusion stochastic partial differential equation (SPDE) (Alexander et al., 2002, 2005). An SPDE implicitly includes the variance structure required to successfully couple not just the mean-level behaviour, but also the variance. The main hurdle with the SPDE approach is how to define the covariance function in order for the variance to be consistent across the entire domain (Kang and Erban, 2019). There is also an open question as to how to incorporate the SPDE approach onto a uniformly growing domain, and obtaining the correct covariance function in that situation.

Domain partitioning and interface position

One key drawback of many spatially extended hybrid methods is that they employ a static interface. This means that once the domain has been partitioned initially, the modelling regime of a region of space remains fixed. In some cases, this is a desirable property: when a fixed region of space needs to be modelled in more detail, such as when considering the behaviour of calcium ions around ion channels on the endoplasmic reticulum (Dobramysl et al., 2015); or when there is *a priori* knowledge that the particle densities in each of the subdomains will not change significantly through time, rendering the chosen representation valid through time. However, these two cases fail to capture the vast majority of possible applications of spatially extended hybrid methods. In many cases, the evolution of the state of a system may be unknown, so it would be advantageous for the method to initially partition the domain, and to dynamically move, create and remove interfaces depending on particle density.

Robinson et al. (2014) utilise an adaptive interface in their "adaptive two regime method", in which the interface moves based on local density. An upper and lower threshold are set. If particle numbers in a region close to the interface in the coarser subdomain were to drop below the lower threshold, that region would be converted to the finer-scale representation. Similarly, if the number of particles in a region adjacent to the interface in the finer subdomain were to rise above the upper threshold, that region would be converted to the coarser representation. The main question here is

how to decide the regions adjacent to the interface in which to count the particles, particularly in the growing domain case due to the difference between Eulerian and Lagrangian coordinate systems. Limited work has also been done with regard the initialisation of the partitioning. Spill et al. (2015) and de la Cruz et al. (2017) each utilise a single threshold, and place the interface in the last compartment which contains sufficient particles to exceed this threshold. The end goal of the initialisation and incorporation of adaptive interfaces would be an algorithm that is able to dynamically create and concatenate new subdomains depending on particle density. For example, if the dynamics dictate that the coarse subdomain has a small number of particles in the middle, it would be better for that region to be modelled by the finer representation. Similarly, if there are two coarse subdomains surrounding a fine subdomain, and the number of particles is high enough in the finer subdomain, then the two coarser regions should be concatenated to become one.

Other hybrid types

Potential avenues for future development include the resolution of the two problems above that arise in spatially extended hybrid methods in the literature. In this section, we briefly describe how other forms of hybrid method may be able to overcome some of these shortcomings, yet still be classed as spatial.

The first is the blending method (Yates et al., 2020), which is described in the introduction chapter. This method could, in theory, be extended to have the blending region cover the entire domain, with the two blending functions changing adaptively depending on the underlying density profile. For example, in regions of lower particle density, the blending function for the coarser paradigm would be lower than that of the finer paradigm, meaning that diffusion occurs more often through the finer regime than the coarser one.

A second type of hybrid method that would alleviate the problems of variance and adaptive subdomains is one in which individual particles are associated with a particular modelling paradigm, and are able to actively switch between representations through the addition of extra "reactions" that are separate from the system kinetics. The switching behaviour would again be related to the relative abundance of particles in the region of space which they occupy.

Neither of these two methods have an interface associated with them, meaning that the issue of choosing where interfaces should go and how they can be created and removed

does not need to be considered. Further, since at least one of the two modelling regimes would be stochastic, and each modelling paradigm is employed across the entire domain, problems with incorrect variance across the domain could be ameliorated.

Realistic domains and applications

Finally, we discuss the extension of hybrid methods to more realistic domains, and moreover, to their applications in the life sciences. We discuss the need for extension to domains with higher dimensions and with more varied geometries, and how this is of paramount importance to being able to represent real-world phenomena.

Most hybrid methods are developed on one-dimensional domains in order to remove as many potential sources of error within the equivalence framework as possible, which allows for a more accurate assessment of the performance of the method. However, for most practical applications, higher dimensions will be required. Many methods, such as the PCM, GCM and ARM, have a natural extension to higher dimensional domains that are defined on (hyper-)cuboids, in which the interface becomes (hyper-)planar. A simple example on the implementation of a hybrid method on a three dimensional domain can be found in Section 4.4 of (Smith and Yates, 2018). However it remains an open question as to how to extend these methods to other, more challenging domain geometries, which will have a direct impact on the applicability to real-world examples.

As mentioned in the discussion of Chapter 2, we hope that the review paper contained therein will highlight hybrid methods to the mathematical and biological communities, allowing for their increased use in applications. However, the lack of examples on two and three dimensions is potentially prohibitive for the development of hybrid methods for real-world applications. There are a few examples of hybrid methods being employed in biology, such as (Dobramysl et al., 2015) who model the formation of "calcium puffs" using the two regime method (Flegg et al., 2012).

Another biological example we have identified is the investigation of the endoplasmic reticulum (ER) refilling mechanism within cells. McIvor et al. (2018) use a pure PDE model in order to investigate the effect of the placement of calcium ion channels on the refilling of the ER. However, in order to achieve analytically tractable solutions, the authors are only able to place ion channels in a circular arrangement, whereas in reality they are placed in a more stochastic distribution. Ion channels opening and closing is also induced by calcium binding, something that can only be accurately resolved by following individual particle locations. We believe that this system would benefit from

a hybrid approach; close to the ion channels, the microscale would be employed to capture the stochastic binding and unbinding of calcium ions, whereas further away, a cheaper method would be utilised where more detail is not as important. There are many such examples in the life sciences that would benefit from applying a hybrid method.

7.2 Final summary

In this thesis, we have developed three spatially extended hybrid methods on growing domains. In order to achieve this, we have further created a macroscopic-to-microscopic method on a static domain and subsequently two equivalence frameworks for the growth of the mesoscale, compartment-based method, and stochastic representations of general Robin boundary conditions on growing domains. We have demonstrated that these methods are accurate, and believe that they will be a valuable tool in years to come, particularly with the continual increase in computing power, and the need for increasingly accurate representations of complex dynamical systems, particularly in the life sciences.

Bibliography

Alexander, F., Garcia, A. and Tartakovsky, D. (2002), 'Algorithm refinement for stochastic partial differential equations: I. Linear diffusion', *J. Comput. Phys.* **182**(1), 47–66.

Alexander, F., Garcia, A. and Tartakovsky, D. (2005), 'Algorithm refinement for stochastic partial differential equations: II. Correlated systems', *J. Comput. Phys.* **207**(2), 769–787.

Andasari, V., Roper, R., Swat, M. and Chaplain, M. (2012), 'Integrating intracellular dynamics using CompuCell3D and Bionetsolver: applications to multiscale modelling of cancer cell growth and invasion', *PLoS One* **7**(3), e33726.

Anderson, D. (2007), 'A modified next reaction method for simulating chemical systems with time dependent propensities and delays', *J. Chem. Phys.* **127**(21), 214107.

Auger, A., Chatelain, P. and Koumoutsakos, P. (2006), 'R-leaping: Accelerating the stochastic simulation algorithm by reaction leaps', *J. Chem. Phys.* **125**, 084103.

Baker, R., Yates, C. and Erban, R. (2010), 'From microscopic to macroscopic descriptions of cell migration on growing domains', *Bull. Math. Biol.* **72**(3), 719–762.

Cianci, C., Smith, S. and Grima, R. (2017), 'Capturing brownian dynamics with an on-lattice model of hard-sphere diffusion', *Phys. Rev. E* **95**(5), 052118.

Crampin, E., Gaffney, E. and Maini, P. (1999), 'Reaction and diffusion on growing domains: scenarios for robust pattern formation', *Bull. Math. Biol.* **61**(6), 1093–1120.

de la Cruz, R., Guerrero, P., Calvo, J. and Alarcón, T. (2017), 'Coarse-graining and hybrid methods for efficient simulation of stochastic multi-scale models of tumour growth', *J. Comput. Phys.* **350**, 974–991.

Deeming, D. and Ferguson, M. (1990), 'Morphometric analysis of embryonic development in alligator mississippiensis, crocodylus johnstoni and crocodylus porosus', *J. Zool.* **221**(3), 419–439.

Dobramysl, U., Rüdiger, S. and Erban, R. (2015), 'Particle-based multiscale modeling of intracellular calcium dynamics', *Multiscale Modeling & Simulation* **14**(3), 997–1016.

Elf, J. and Ehrenberg, M. (2004), 'Spontaneous separation of bi-stable biochemical systems into spatial domains of opposite phases.', *Syst. Biol.* **1**(2), 230–236.

Erban, R. and Chapman, S. (2007), 'Reactive boundary conditions for stochastic simulations of reaction–diffusion processes', *Phys. Biol.* **4**(1), 16–28.

Erban, R. and Chapman, S. (2009), 'Stochastic modelling of reaction–diffusion processes: algorithms for bimolecular reactions', *Phys. Biol.* **6**(4), 1–18.

Flegg, M., Chapman, S. and Erban, R. (2012), 'The two-regime method for optimizing stochastic reaction–diffusion simulations', *J. Roy. Soc. Interface.* **9**(70), 859–868.

Flegg, M., Hellander, S. and Erban, R. (2015), 'Convergence of methods for coupling of microscopic and mesoscopic reaction-diffusion simulations', *J. Comput. Phys.* **289**(C), 1–17.

Flekkøy, E., Feder, J. and Wagner, G. (2001), 'Coupling particles and fields in a diffusive hybrid model', *Phys. Rev. E* **64**(6), 066302.

Franz, B., Flegg, M., Chapman, S. and Erban, R. (2013), 'Multiscale reaction-diffusion algorithms: PDE-assisted Brownian dynamics', *SIAM J. Appl. Math.* **73**(3), 1224–1247.

Gibson, M. and Bruck, J. (2000), 'Efficient exact stochastic simulation of chemical systems with many species and many channels', *J. Phys. Chem. A.* **104**(9), 1876–1889.

Gillespie, D. (1977), 'Exact stochastic simulation of coupled chemical reactions', *J. Phys. Chem.* **81**(25), 2340–2361.

Gillespie, D. (2001), 'Approximate accelerated stochastic simulation of chemically reacting systems', *J. Chem. Phys.* **115**(4), 1716–1733.

Gorba, C., Geyer, T. and Helms, V. (2004), 'Brownian dynamics simulations of simplified cytochrome c molecules in the presence of a charged surface', *J. Chem. Phys.* **121**(1), 457–464.

Greenhalgh, D. (1998), 'The role of apoptosis in wound healing', *Int. J. Biochem. Cell Biol.* **30**(9), 1019–1030.

Grinnell, F., Zhu, M., Carlson, M. and Abrams, J. (1999), 'Release of mechanical tension triggers apoptosis of human fibroblasts in a model of regressing granulation tissue', *Exp. Cell. Res.* **248**(2), 608–619.

Harrison, J. and Yates, C. (2016), 'A hybrid algorithm for coupling partial differential equation and compartment-based dynamics', *J. Roy. Soc. Interface.* **13**(122), 20160335.

Isaacson, S. (2013), 'A convergent reaction-diffusion master equation', *J. Chem. Phys.* **139**(5), 054101.

Kang, H. and Erban, R. (2019), 'Multiscale stochastic reaction–diffusion algorithms combining markov chain models with stochastic partial differential equations', *Bull. Math. Biol.* **81**(8), 3185–3213.

Khan, S., Zou, Y., Amjad, A., Gardezi, A., Smith, C., Winters, C. and Reese, T. (2011), 'Sequestration of CaMKII in dendritic spines in silico', *J. Comput. Neurosci.* **31**(3), 581–594.

Korsgaard, B. and Andersen, F. (1985), 'Embryonic nutrition, growth and energetics inzoarces viviparus l. as indication of a maternal-fetal trophic relationship', *J. Comp. Physiol. B* **155**(4), 437–444.

Kostré, M., Schütte, C., Noé, F. and del Razo, M. (2020), 'Coupling particle-based reaction-diffusion simulations with reservoirs mediated by reaction-diffusion pdes', *arXiv preprint arXiv:2006.00003* .

Kulesa, P., Cruywagen, G., Lubkin, S., Main, P., Sneyd, J., Ferguson, M. and Murray, J. (1996), 'On a model mechanism for the spatial patterning of teeth primordia in the alligator', *J. Theor. Biol.* **180**(4), 287–296.

Leshem, A., Ar, A. and Ackerman, R. (1991), 'Growth, water, and energy metabolism of the soft-shelled turtle (trionyx triunguis) embryo: effects of temperature', *Physiol. Zool.* **64**(2), 568–594.

McIvor, E., Coombes, S. and Thul, R. (2018), 'Three-dimensional spatio-temporal modelling of store operated Ca^{2+} entry: insights into ER refilling and the spatial signature of Ca^{2+} signals', *Cell Calcium* **73**, 11–24.

McLennan, R., Dyson, L., Prather, K., Morrison, J., Baker, R., Maini, P. and P.M., K. (2012), 'Multiscale mechanisms of cell migration during development: theory and experiment', *Development* **139**(16), 2935–2944.

Moro, E. (2004), 'Hybrid method for simulating front propagation in reaction-diffusion systems', *Phys. Rev. E* **69**(6), 060101.

Murray, J., Deeming, D. and Ferguson, M. (1990), 'Size-dependent pigmentation-pattern formation in embryos of alligator mississippiensis: time of initiation of pattern generation mechanism', *Proc. Royal Soc. B* **239**(1296), 279–293.

Risken, H. (1996), Fokker-planck equation, *in* 'The Fokker-Planck Equation', Springer, pp. 63–95.

Robinson, M., Flegg, M. and Erban, R. (2014), 'Adaptive two-regime method: application to front propagation', *J. Chem. Phys.* **140**(12), 124109.

Sherratt, J. (2005), 'An analysis of vegetation stripe formation in semi-arid landscapes', *J. Math. Biol.* **51**(2), 183–197.

Simpson, M. (2015), 'Exact solutions of linear reaction-diffusion processes on a uniformly growing domain: Criteria for successful colonization', *PLoS One* **10**(2), e0117949.

Singer, A., Schuss, Z., Osipov, A. and Holcman, D. (2008), 'Partially reflected diffusion', *SIAM J. Appl. Math.* **68**(3), 844–868.

Smith, C., Mailler, C. and Yates, C. (2019), 'Unbiased on-lattice domain growth', *Phys. Rev. E* **100**(6).

Smith, C. and Yates, C. (2018), 'The auxiliary region method: A hybrid method for coupling a PDE to Brownian-based dynamics for reaction-diffusion systems', *R. Soc. Open Sci.* **5**(8).

Smith, G. (1985), *Numerical solution of partial differential equations: finite difference methods*, Oxford University Press.

Smith, S., Cianci, C. and Grima, R. (2016), 'Analytical approximations for spatial stochastic gene expression in single cells and tissues', *J. Roy. Soc. Interface* **13**(118), 20151051.

Spill, F., Guerrero, P., Alarcon, T., Maini, P. and Byrne, H. (2015), 'Hybrid approaches for multiple-species stochastic reaction–diffusion models', *J. Comput. Phys.* **299**, 429–445.

Taylor, P., Baker, R. and Yates, C. (2015), 'Deriving appropriate boundary conditions, and accelerating position-jump simulations, of diffusion using non-local jumping', *Phys. Biol.* **12**(1), 016006.

Tian, T. and Burrage, K. (2004), 'Binomial leap methods for simulating stochastic chemical kinetics', *J. Chem. Phys.* **121**(21), 10356–10364.

Turing, A. (1952), 'The chemical basis of morphogenesis', *Phil. Trans. R. Soc. B.* **237**(641), 37–72.

Volpert, V. and Petrovskii, S. (2009), 'Reaction–diffusion waves in biology', *Phys. Life Rev.* **6**(4), 267–310.

Woolley, T., Baker, R., Gaffney, E. and Maini, P. (2011), 'Stochastic reaction and diffusion on growing domains: understanding the breakdown of robust pattern formation', *Phys. Rev. E* **84**(4), 046216.

Yates, C. (2014), 'Discrete and continuous models for tissue growth and shrinkage', *J. Theor. Biol.* **350**, 37–48.

Yates, C. and Flegg, M. (2015), 'The pseudo-compartment method for coupling partial differential equation and compartment-based models of diffusion', *J. Roy. Soc. Interface* **12**(106), 20150141.

Yates, C., George, A., Jordana, A., Smith, C., Duncan, A. and Zygalakis, K. (2020), 'The blending region hybrid framework for the simulation of stochastic reaction–diffusion processes', *J. Roy. Soc. Interface* **17**(171), 20200563.

ZhuGe, R., Fogarty, K., Tuft, R., Lifshitz, L., Sayar, K. and Walsh, J. (2000), 'Dynamics of signaling between Ca2+ sparks and Ca2+-activated K+ channels studied with a novel image-based method for direct intracellular measurement of ryanodine receptor Ca2+ current', *J. Gen. Physiol.* **116**(6), 845–864.